Electric Vehicle Charging Pattern Prediction

Mariana Martins – Computational Science and Engineering

Weile Weng – Mathematics

Simon Klotz – Data Engineering and Analytics

sonnen

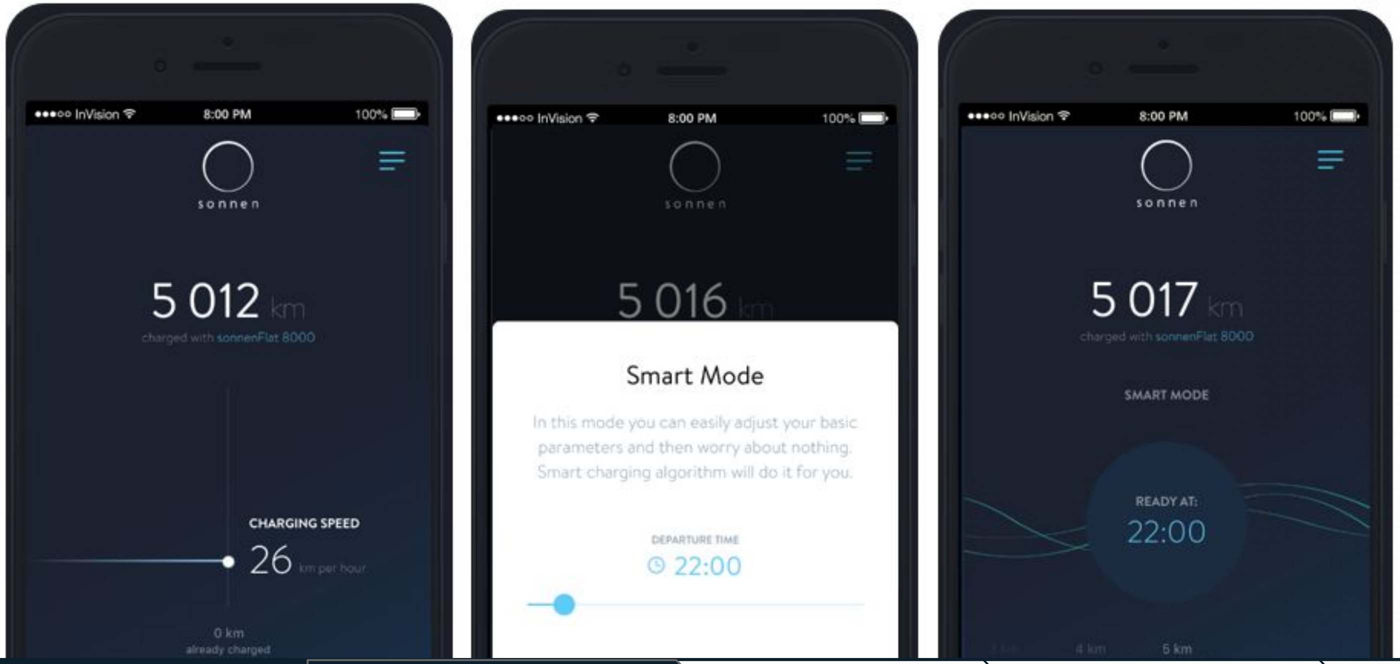
Technical University of Munich, 25.07.2018

Agenda

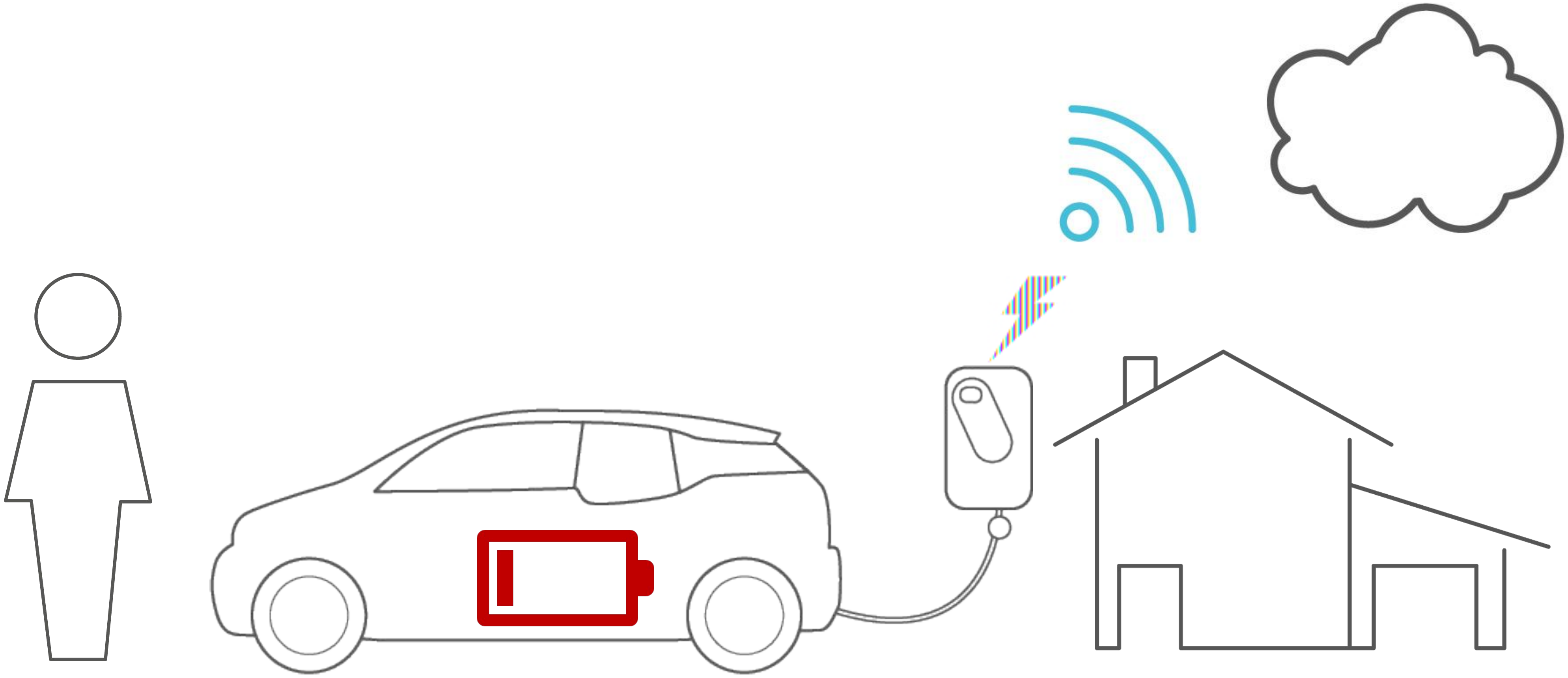
sonnenCharger → Data → Methods → Evaluation → Conclusion

sonnenCharger

- Launched: April 2018
- Smartphone app integration
- Two modes: Power and Smart

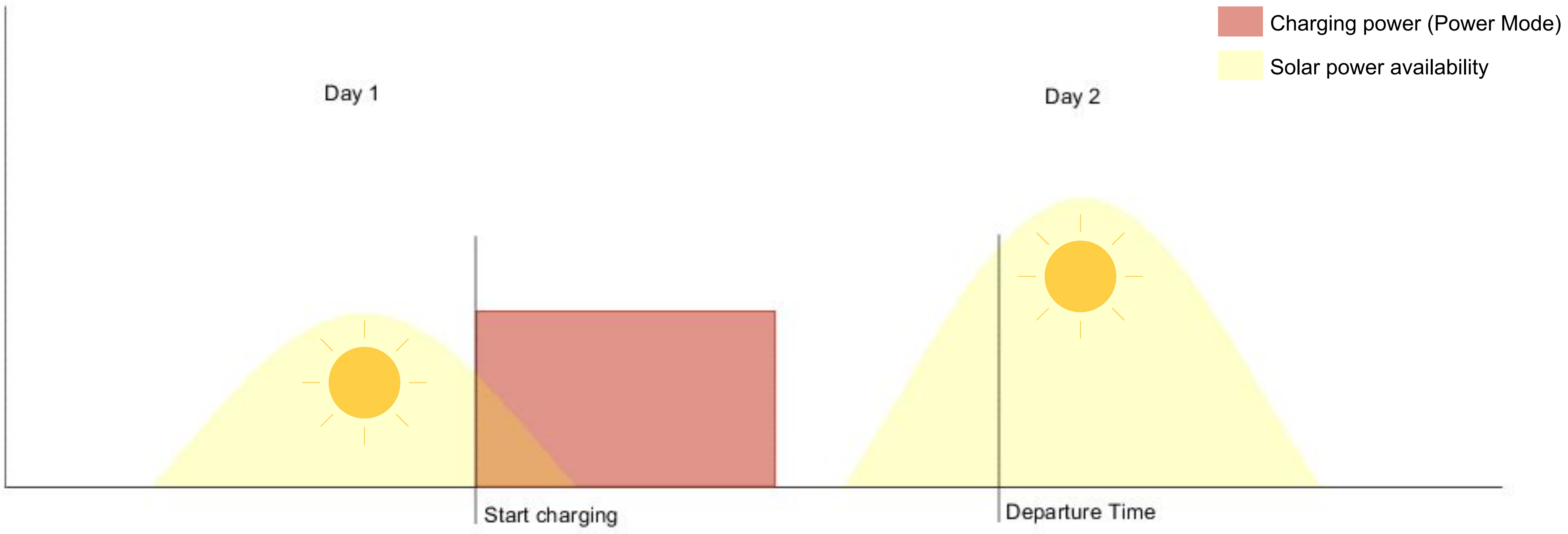


Power Mode



sonnenCharger modes

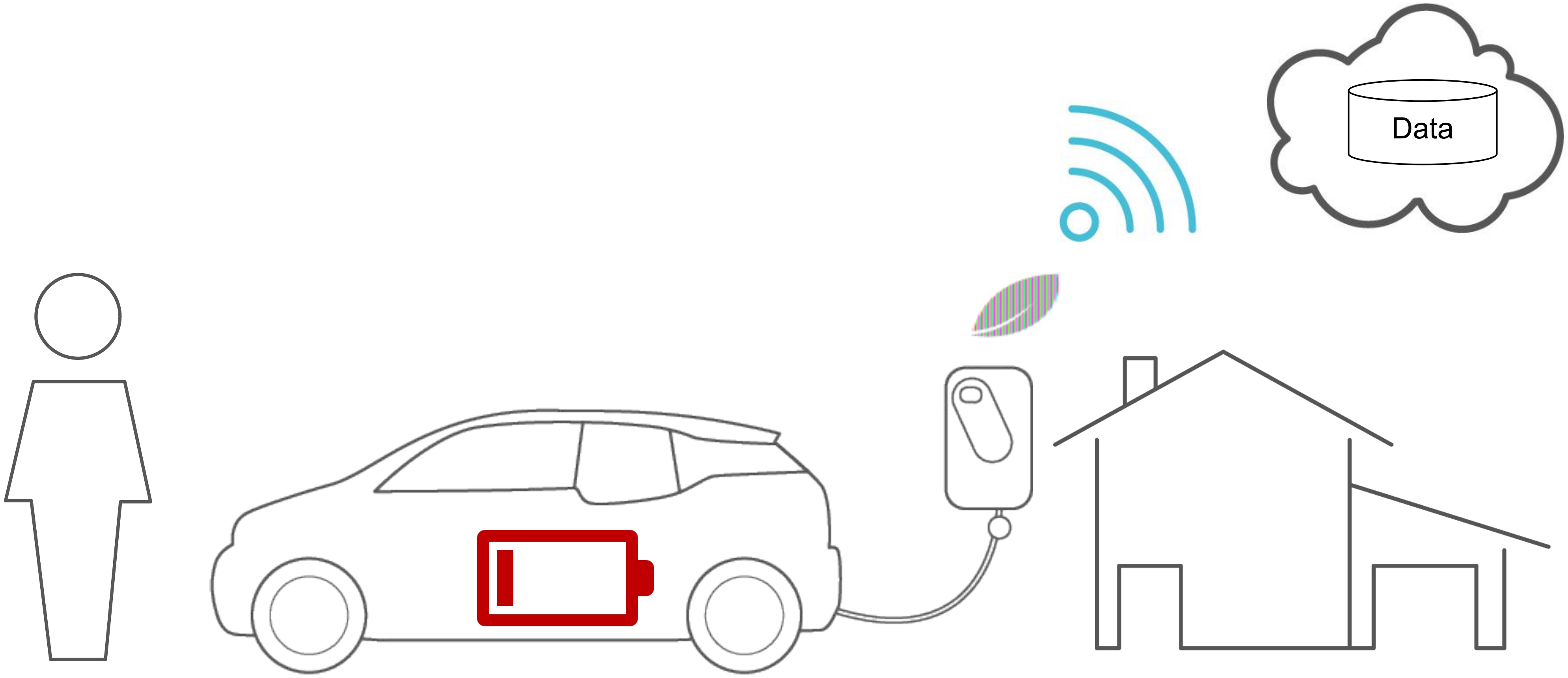
Power Mode



- **Power Mode:** Charge as fast as possible

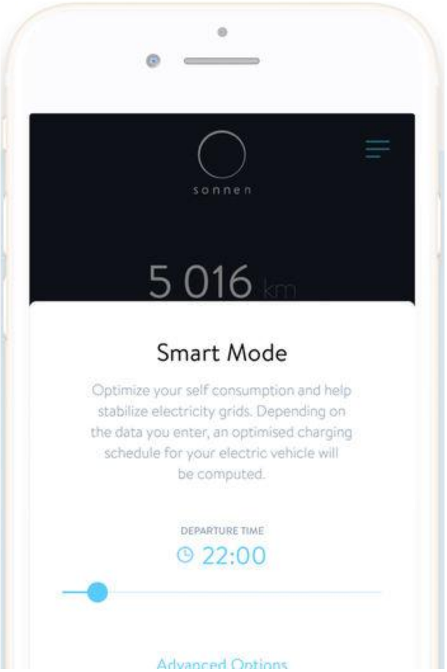
sonnenCharger modes

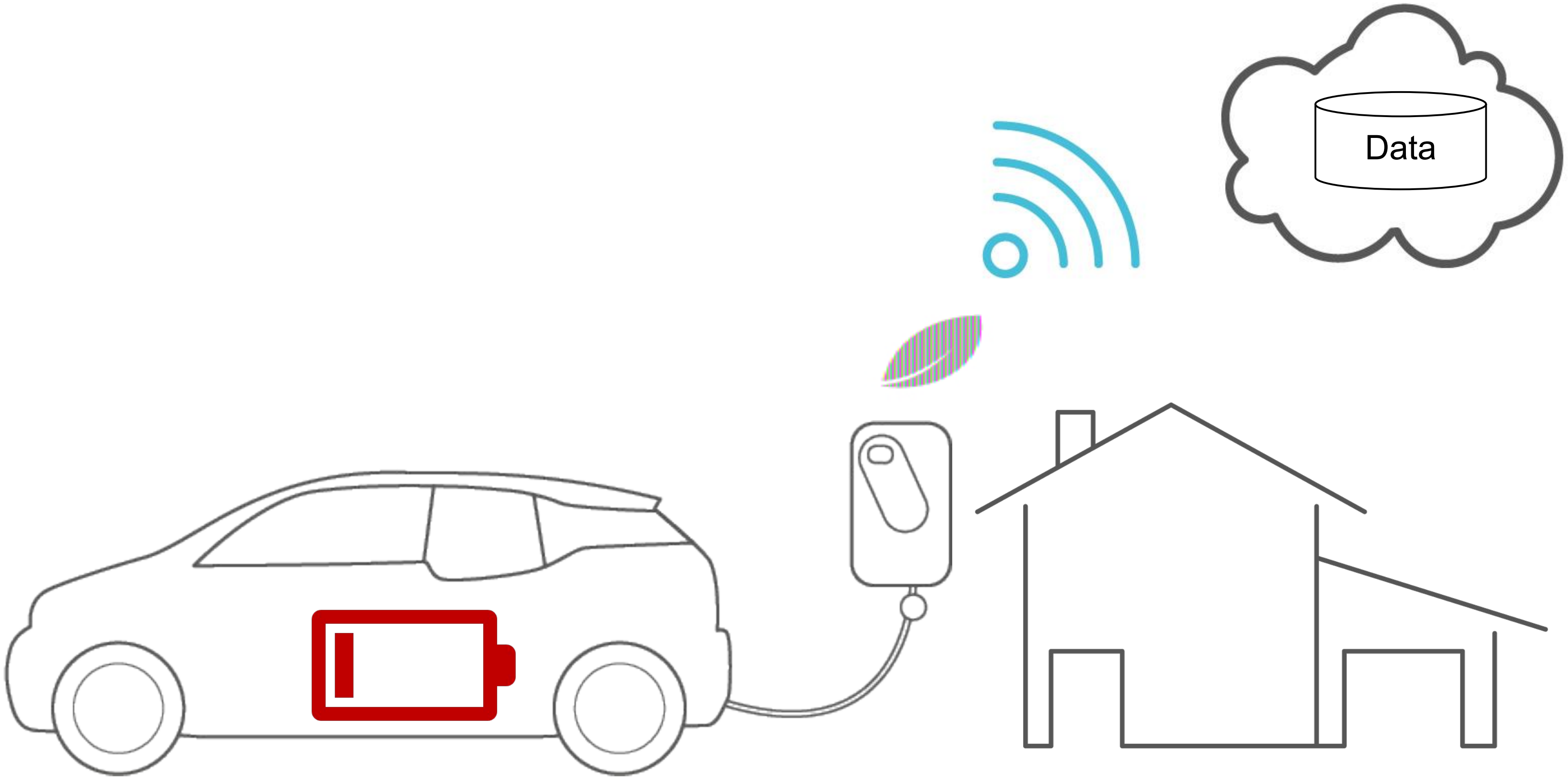
Smart Mode



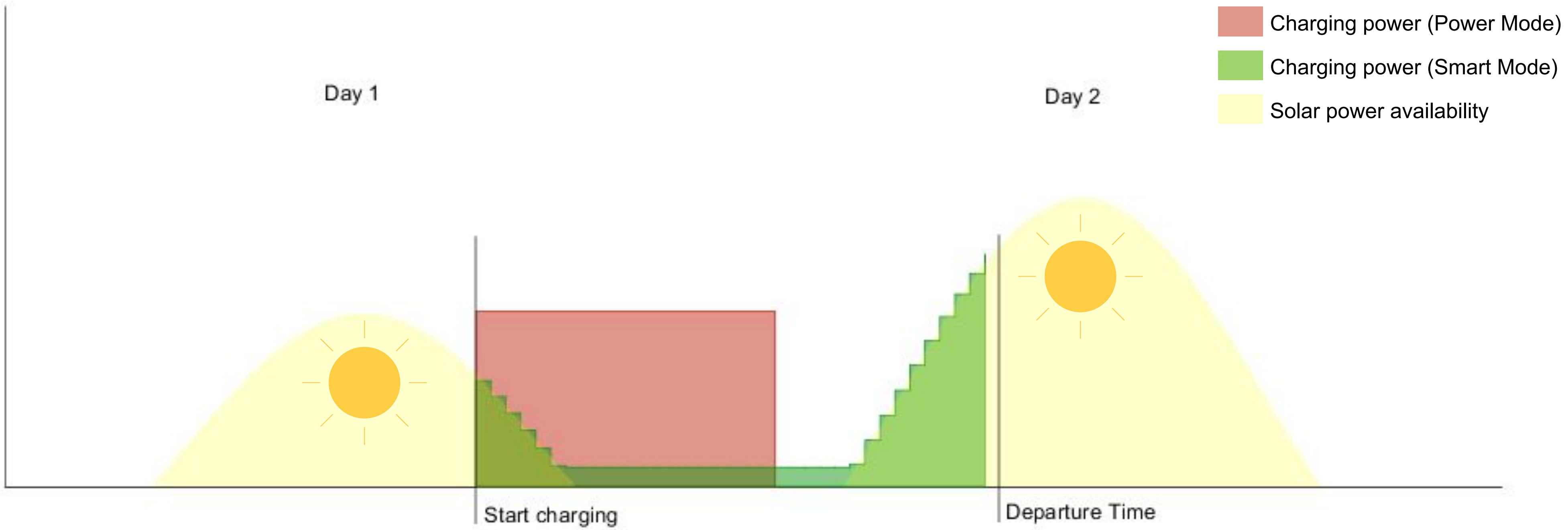
sonnenCharger modes

Smart Mode





Smart Mode

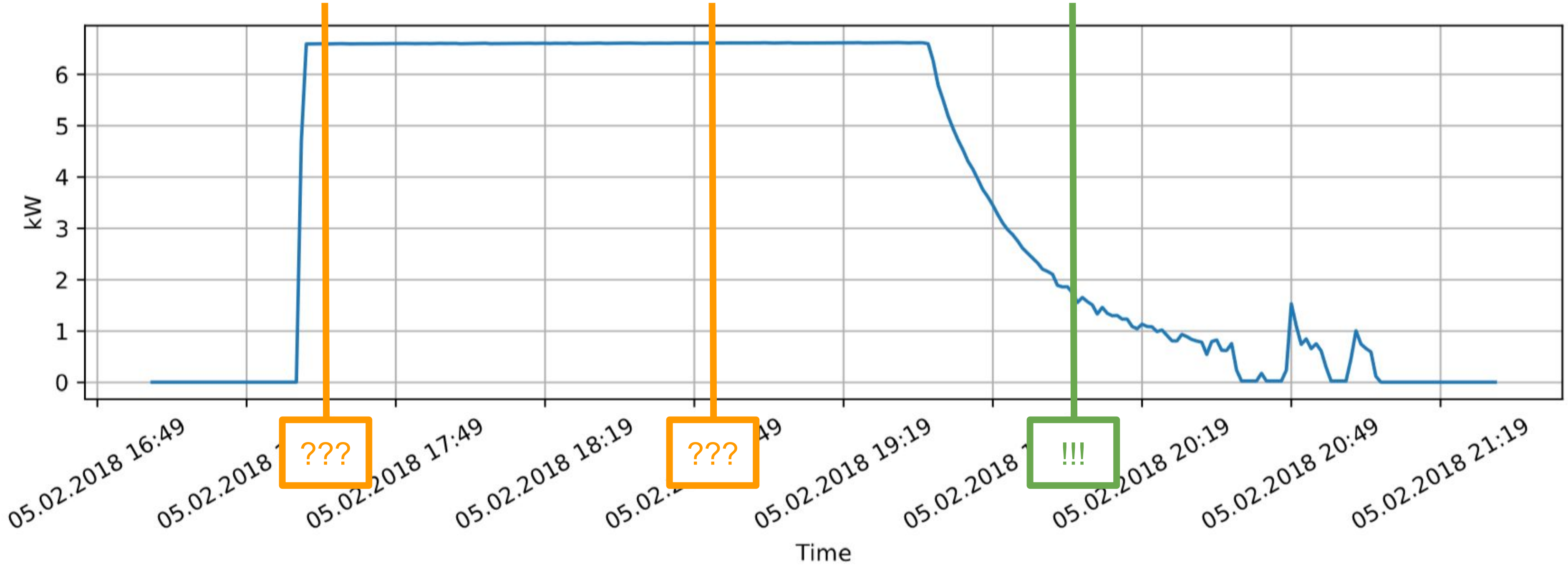


- Power Mode: Charge as fast as possible
- Smart Mode: Maximize use of solar power -- how?

Goal: Estimate required energy

- In order to create a smart charging profile, we must know how much energy the car needs **when it is plugged in**

required energy = battery capacity – current amount of charge



Goal: Estimate required energy

- In order to create a smart charging profile, we must know how much energy the car needs **when it is plugged in**

required energy = *battery capacity* – ***current amount of charge***

- Not enough input data for a physics-based model of the battery
- No interface to get data from the car

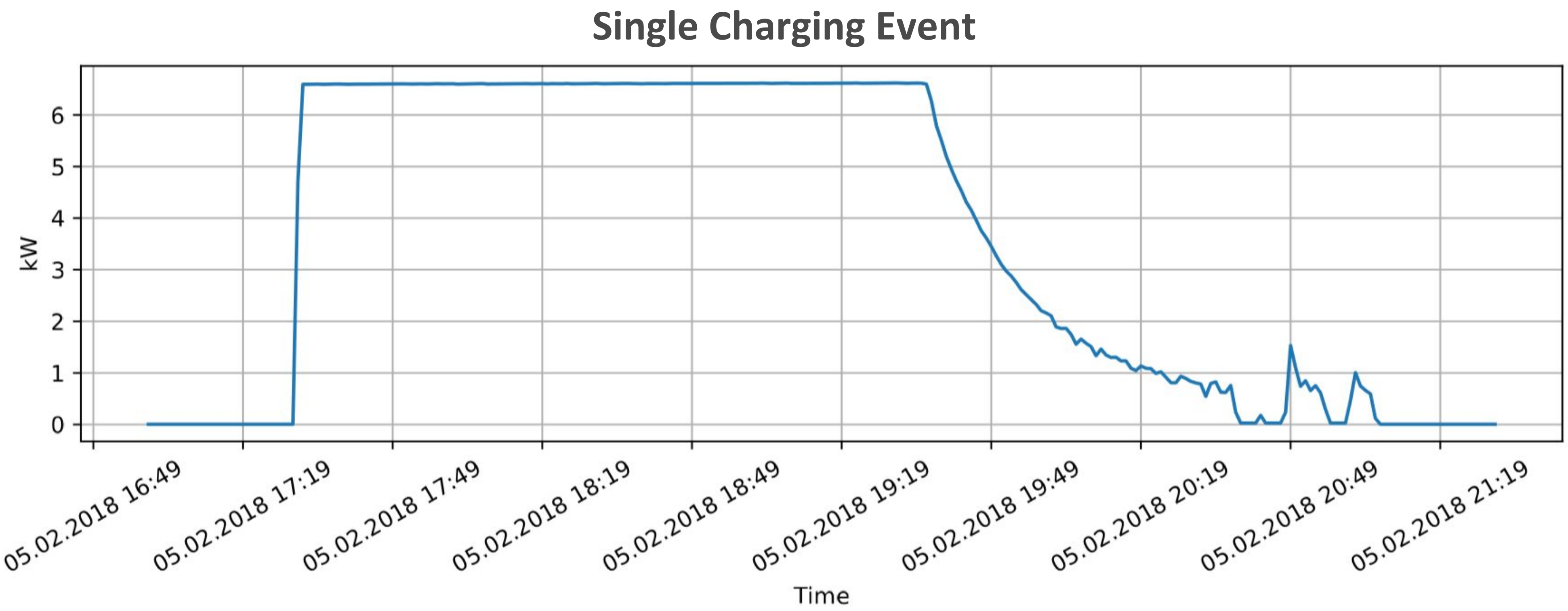
- **Our approach: Estimate based on data from past charging events (usage habits)**

Data source

Dataport

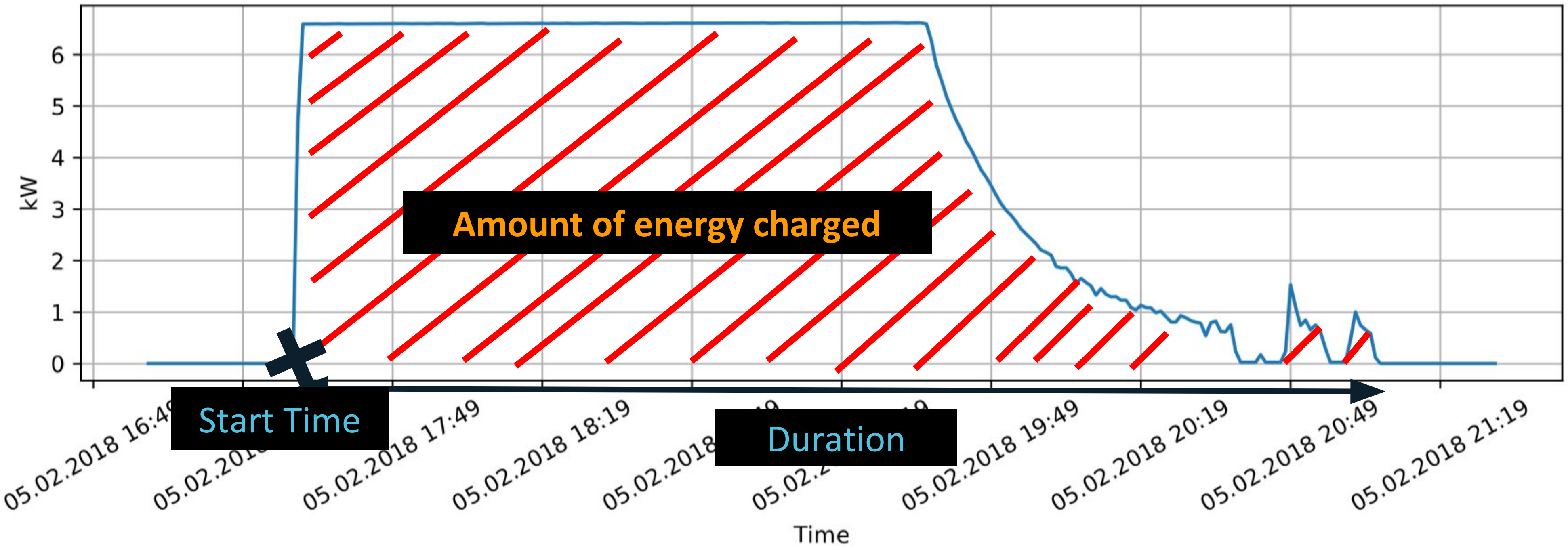
- Dataset: Pecan Street's residential electricity use research
  - Anonymized data from over 1300 volunteers back to 2016
  - Circuit-level (disaggregated) and whole-home electricity use data
  - In particular: electricity usage data for home EV charging
  - Drawback: only power data (no car model information, data on interrupted charging, …)

Feature Extraction



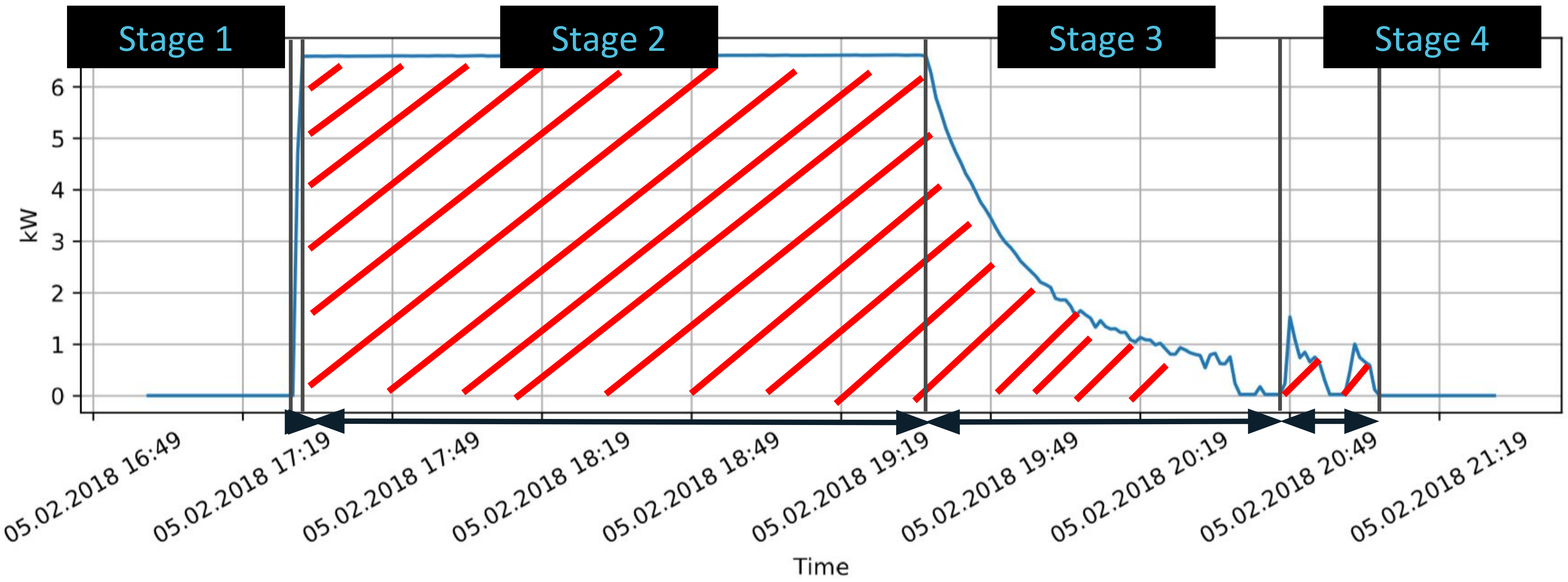
➔ Which aspects of the time series could be relevant for our prediction?

Charging Features



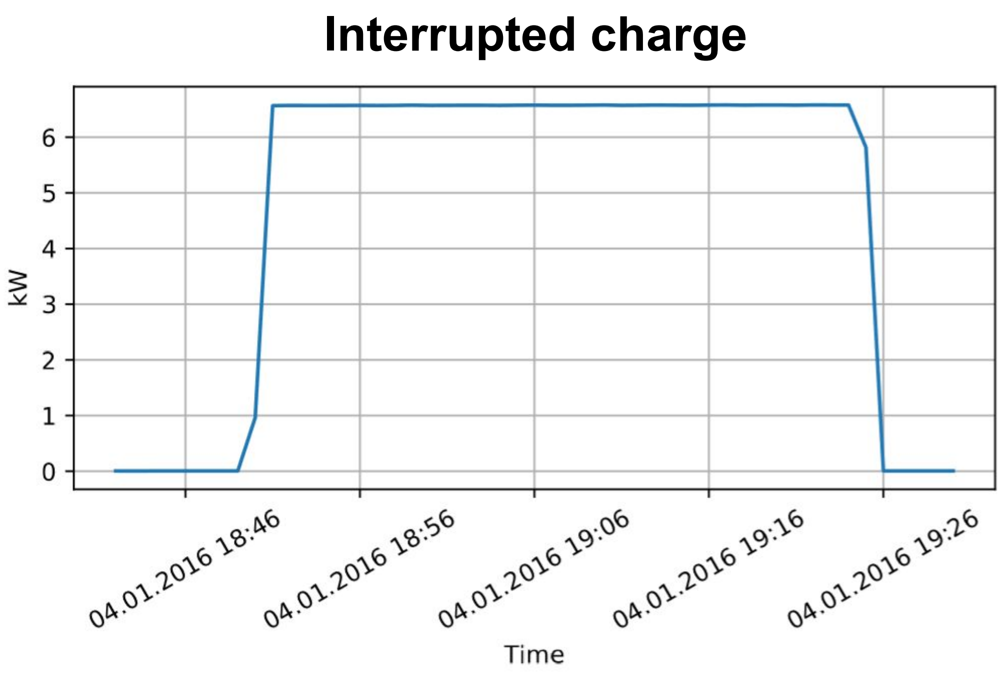
Feature Extraction

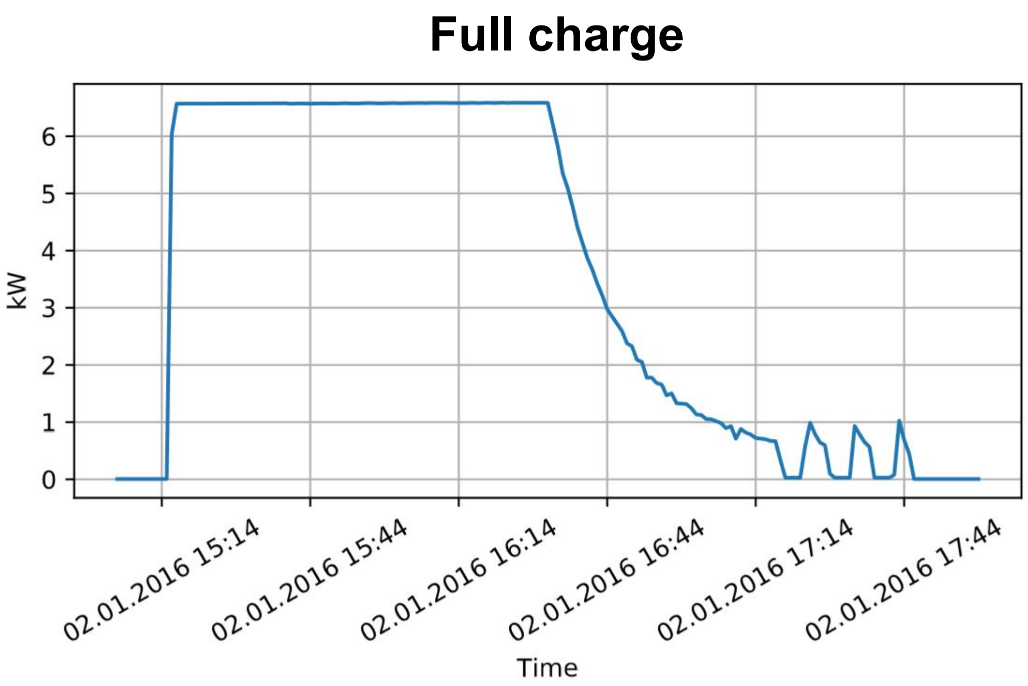
4 Stages



Feature Extraction

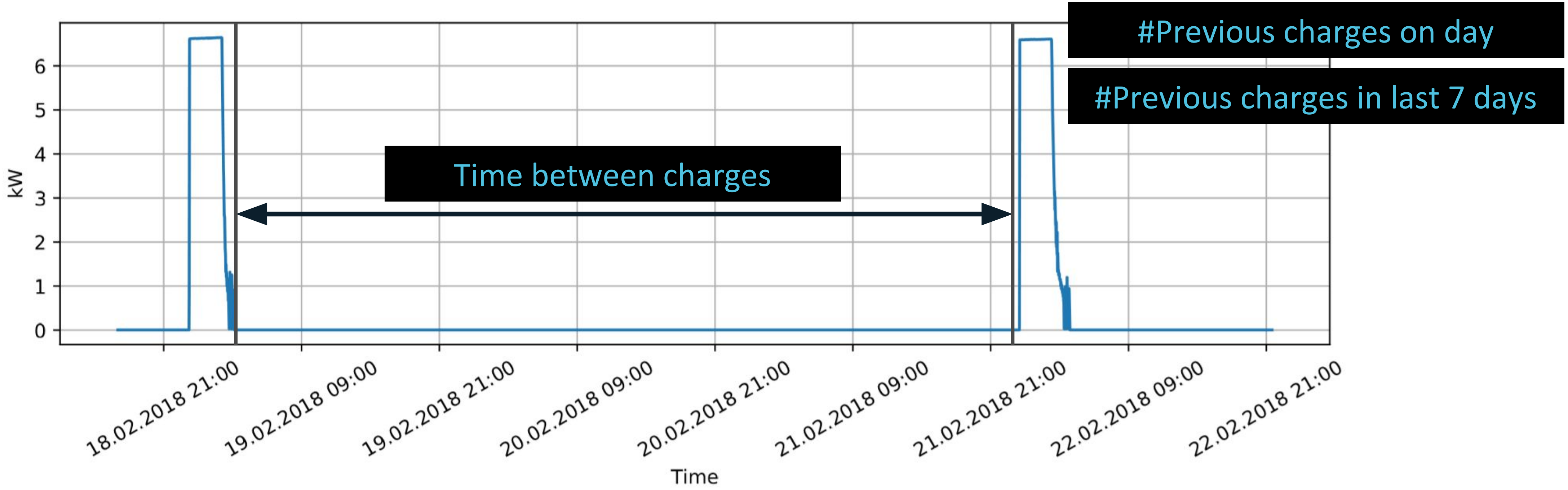
Full Charge





Feature Extraction

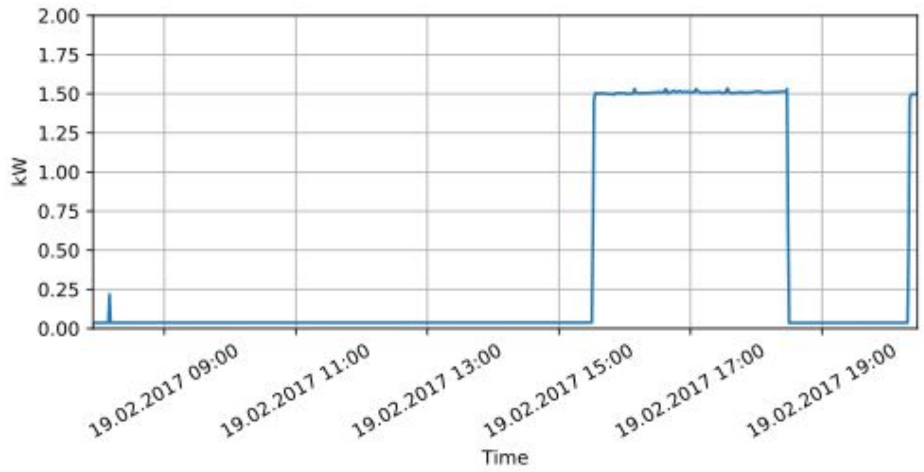
Previous Charges

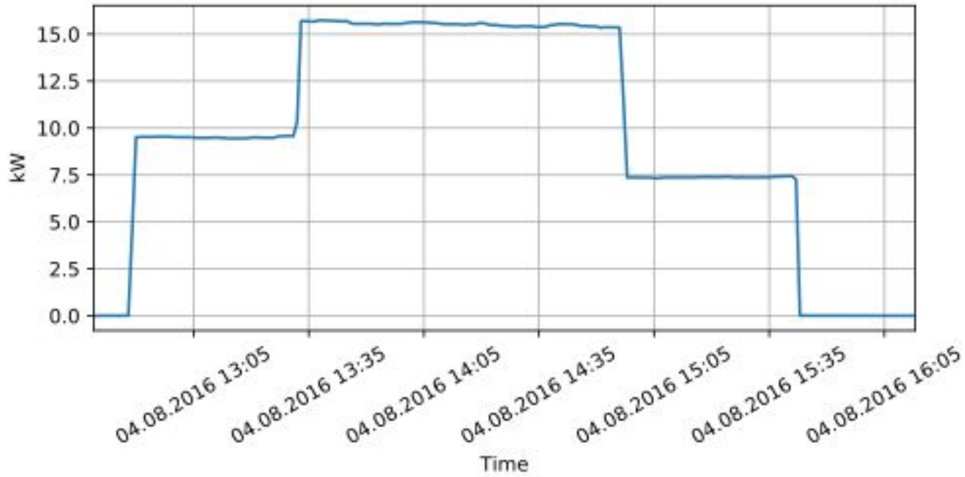


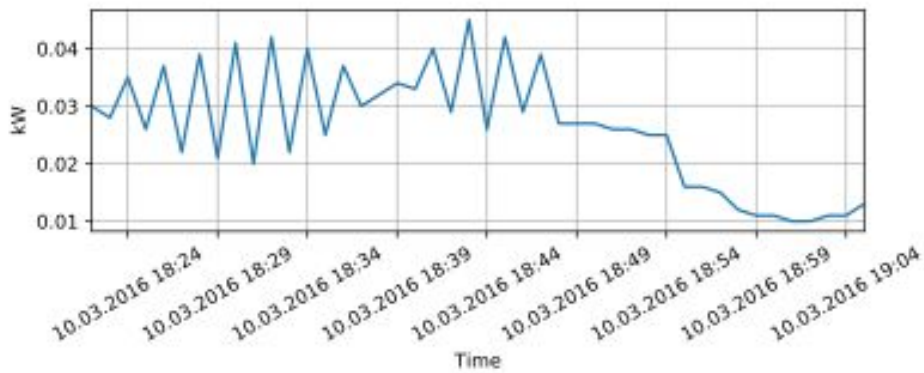
Data cleaning

Exclude

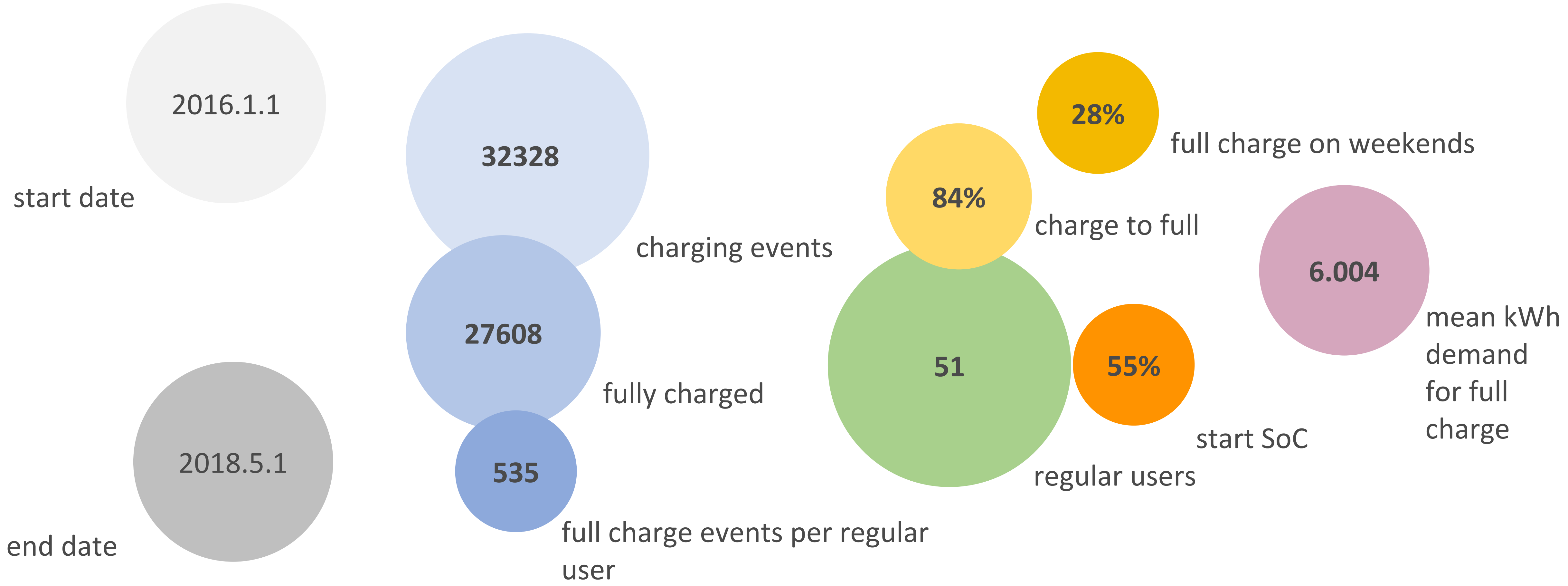
- noisy charging events, with threshold=0.1kW
- charging events that have not reached expected steady state
- only full charges
- users with less than 50 full charges
- users with irregular EV charging profiles





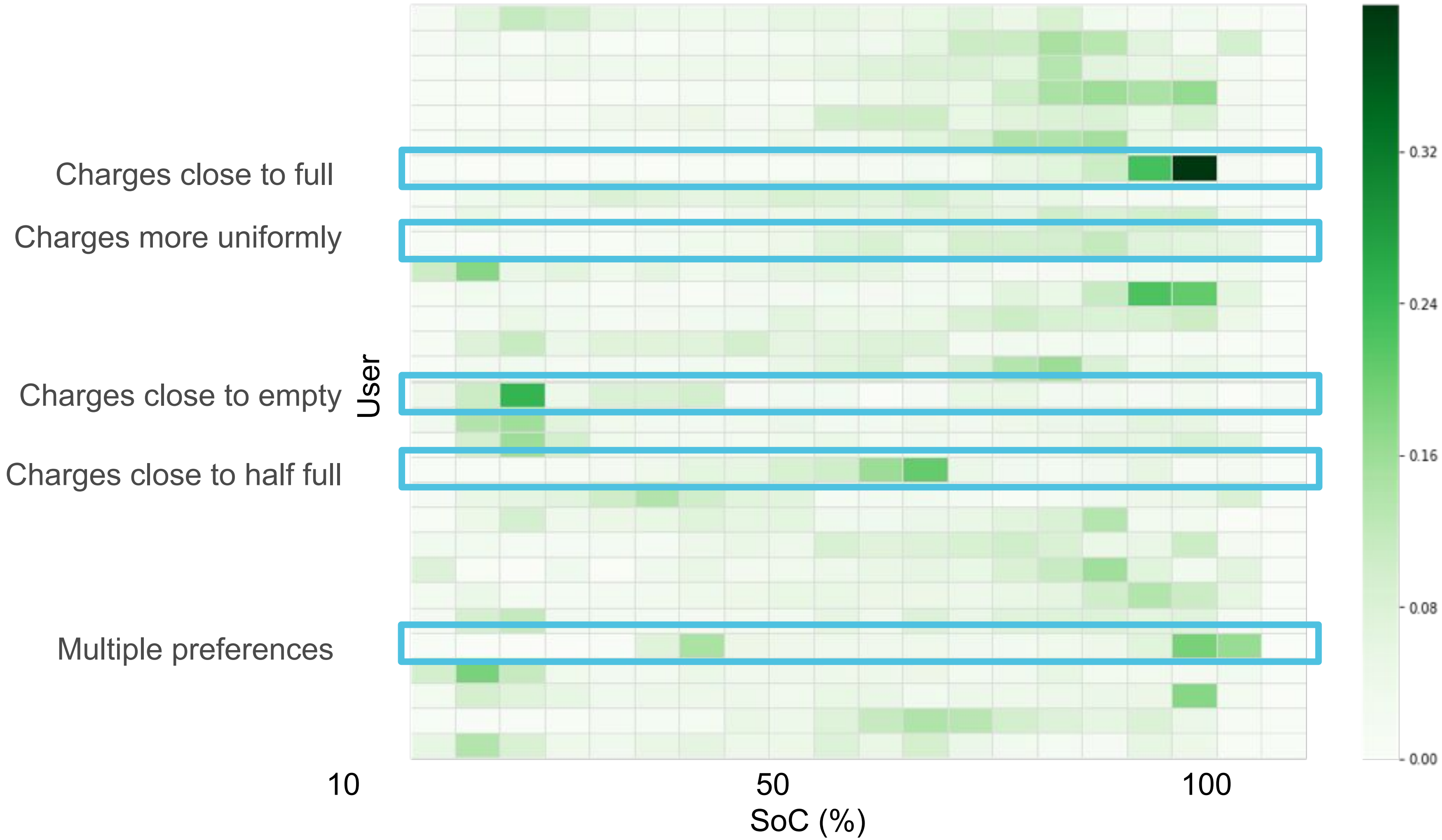


Data Description



Fully charged events of regular users

Distribution of start state of charge (SoCs)



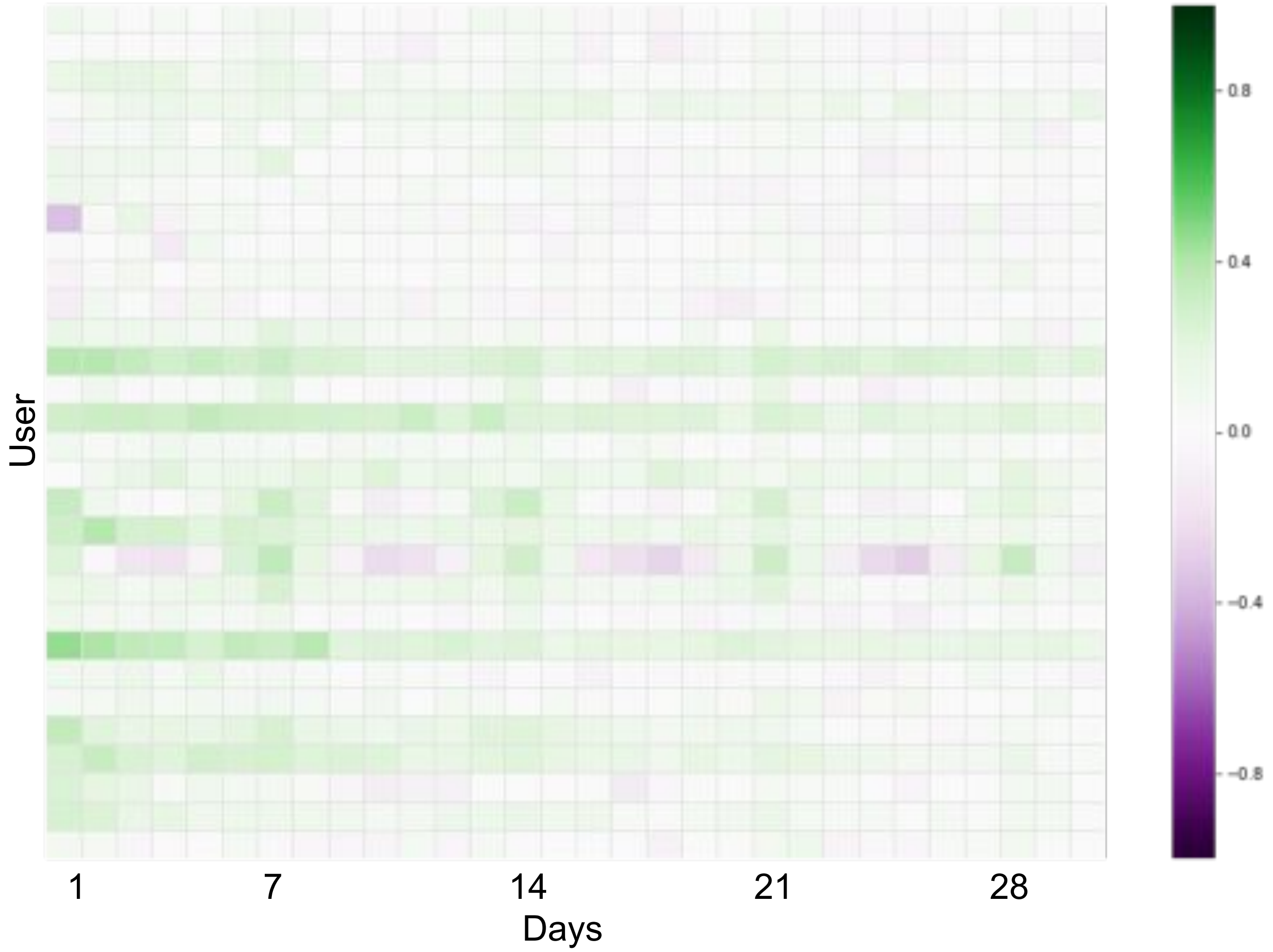
Fully charged events of regular users

Autocorrelation of required energy

Low autocorrelation of required energy over the time series:

- Diverse pattern among different users
- Slightly higher value in same day of week

→ Linear regression would not fit well, with charged energy in the past as input
→ Provide intuition for conditional probabilistic approach

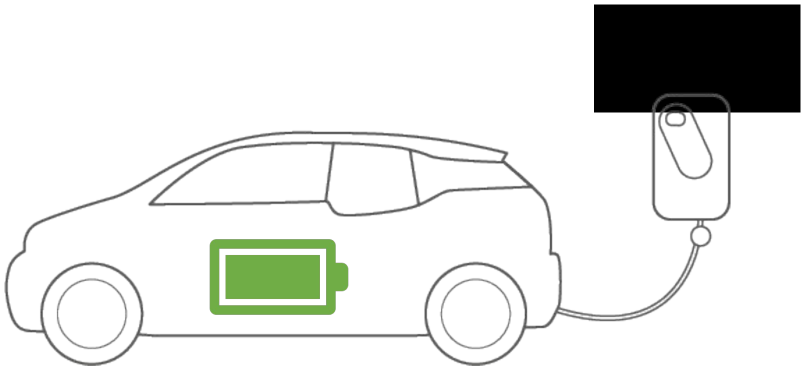


Failure Mode Analysis

Predicted >> Actual

⇩

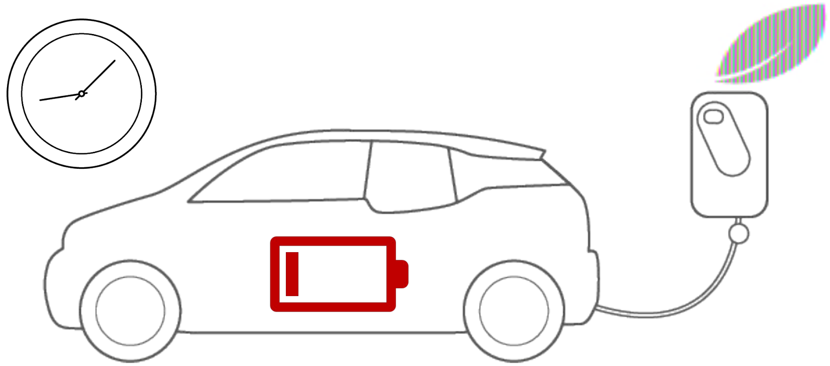
Suboptimal use of clean energy



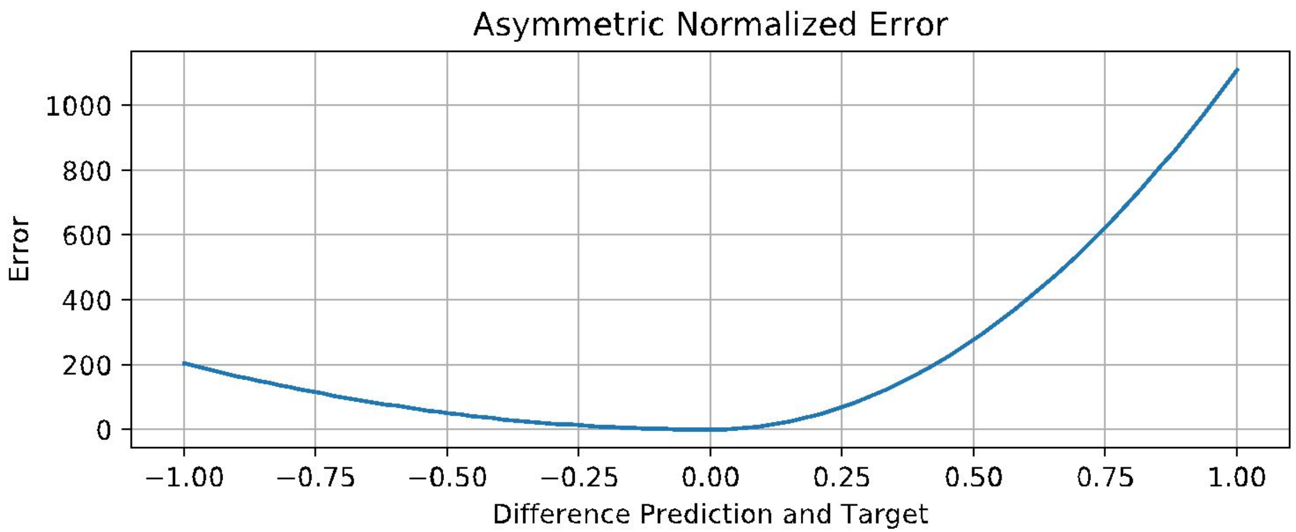
Predicted << Actual

⇩

Not fully charged at departure time



Asymmetric Quadratic Error (AQE)



- Normalize required energy
- Underestimation is penalized more than overestimation

Advantage:
→ Accommodate user needs

Further improvement:
→ customize coefficients conditioned on battery capacity

General Approach

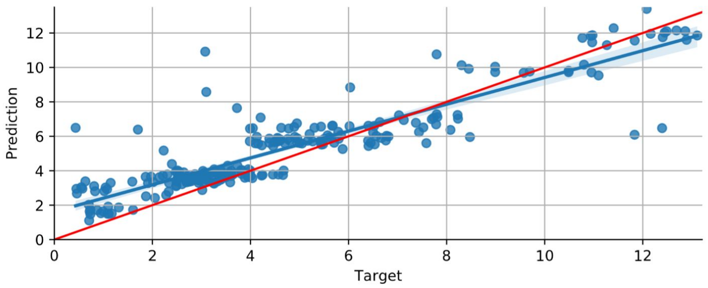
- Data of current charge
  - Start time
  - Time since last charge
  - …
- Past data of last X charges
  - Start time
  - Energy charged
  - ...

Machine Learning Models

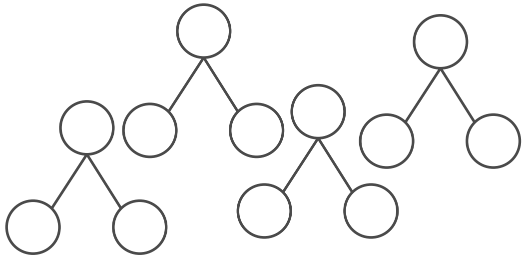
- **Mean Model**
  - Baseline

$$y_{new} = \frac{1}{n}\sum_{i=0}^{n} y_i$$

- **Ridge Regression**
  - Simple model

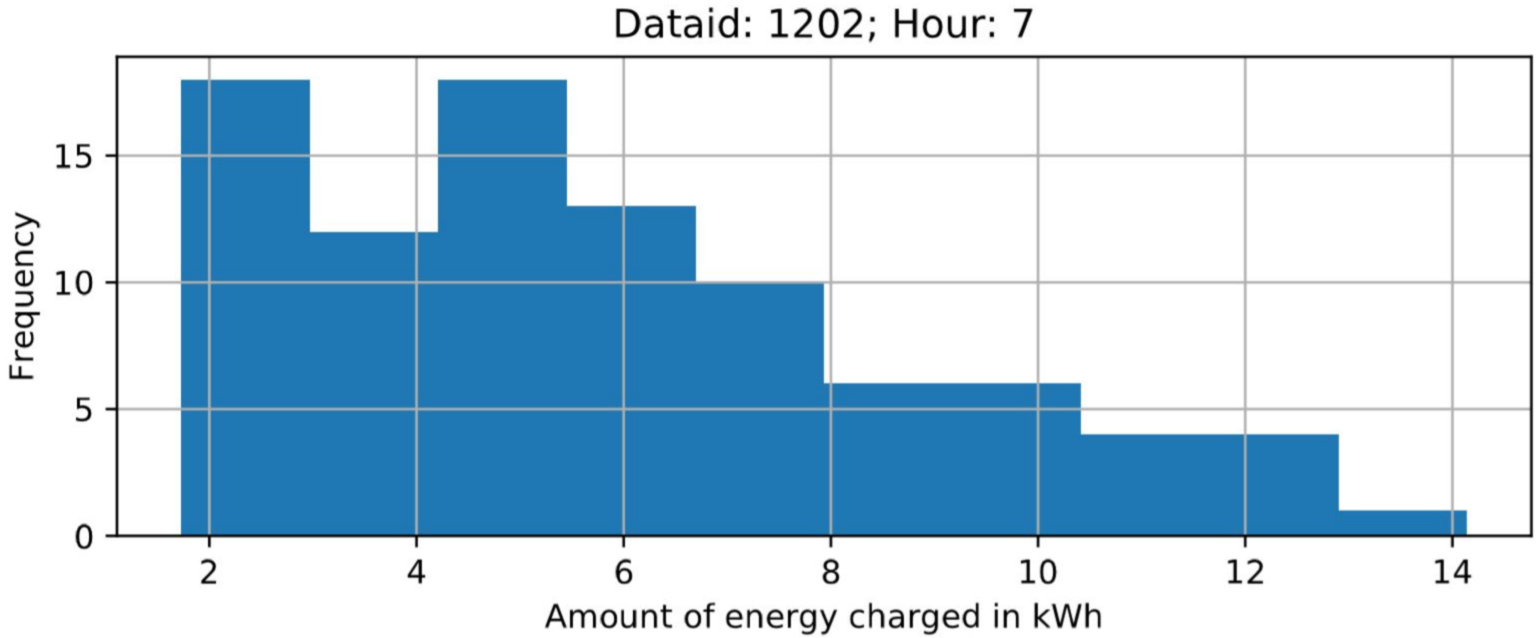


- **XGBoost**
  - Good performance on structured datasets
  - Custom objective function



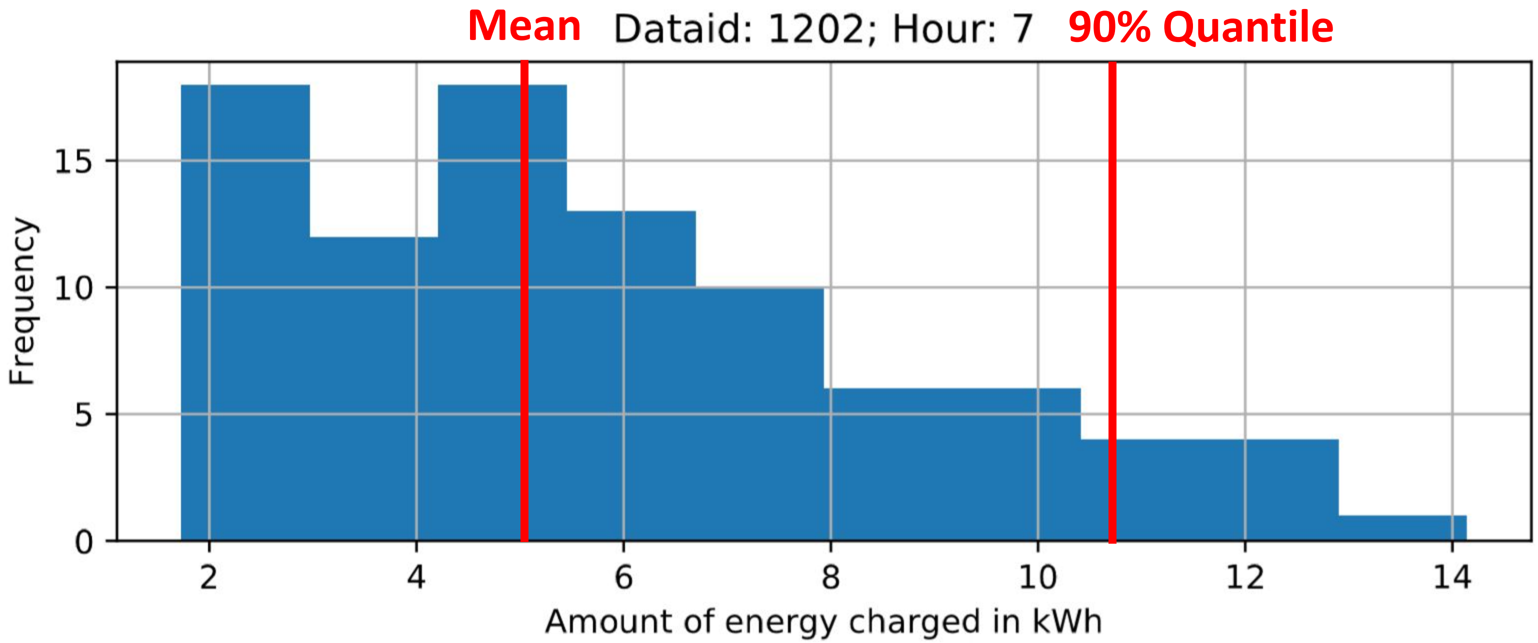
Conditional Probability Approach

- Goal: Estimate whole distribution instead of point predictions
- Only discrete features



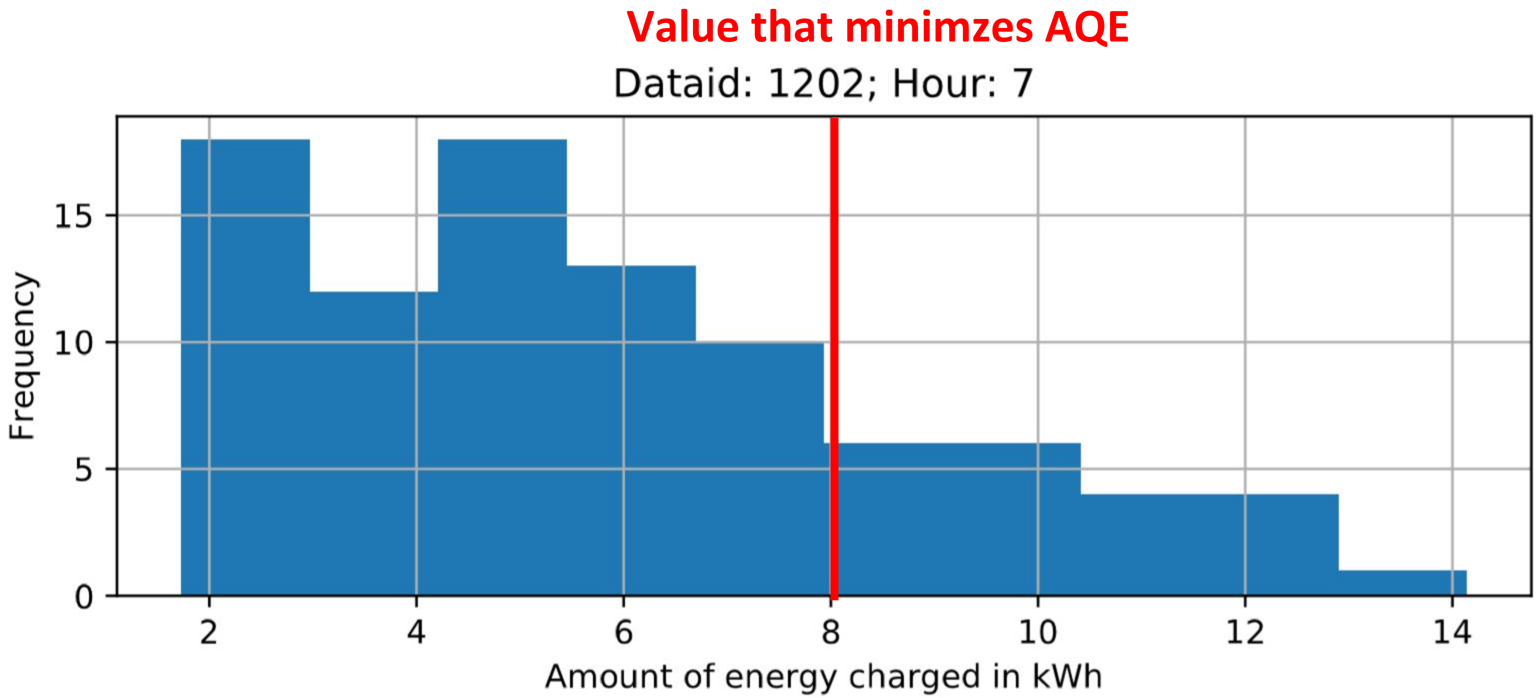
Predictions using Distribution

Different ways to make predictions based on distribution:



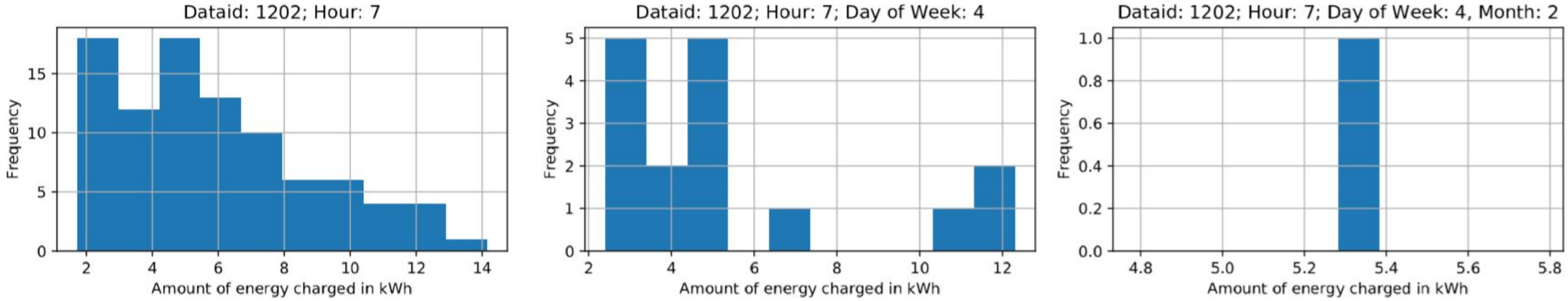
Predictions using Distribution

We use value that minimizes our error function with regard to distribution:



Conditional Probability Approach

Different Granularities



- Too few conditions: Distribution not specific enough
- Too many conditions: Not enough data points available

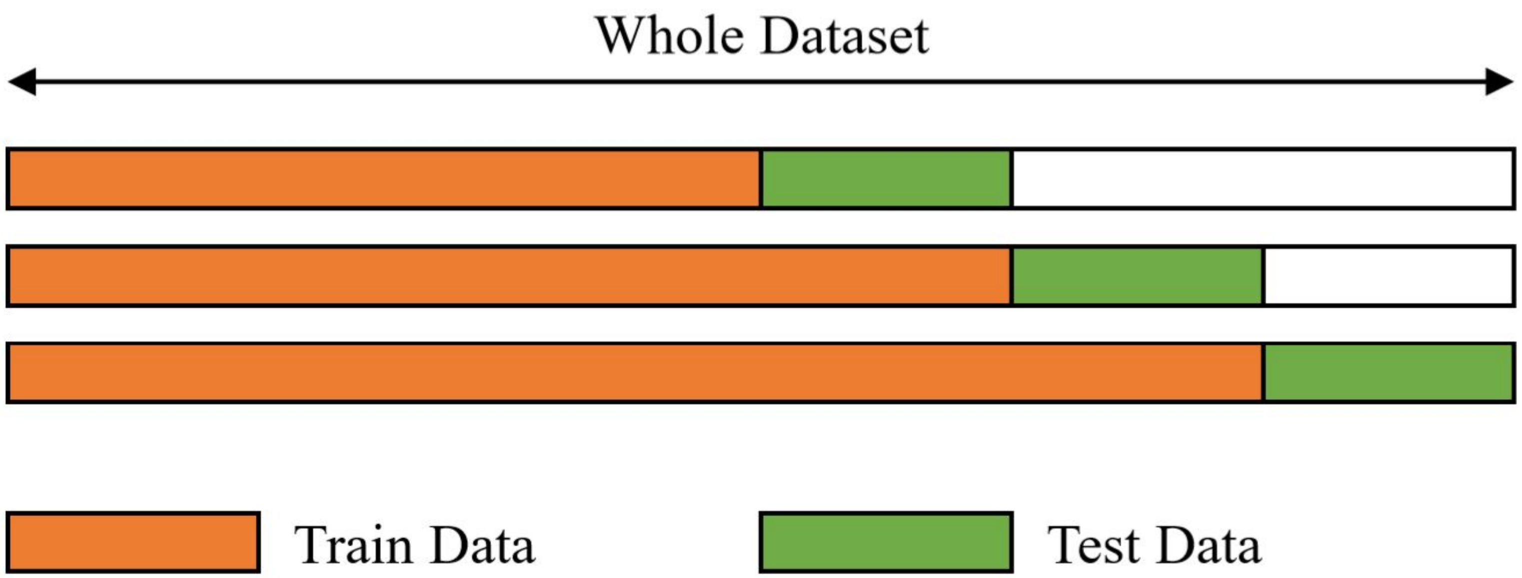
➔ Sequentially drop conditions until threshold of datapoints is reached

➔ hour divided by eight/four, hour, number of previous charges during week, month, day of week, season, number of previous charges during day

Rolling-origin-update evaluation

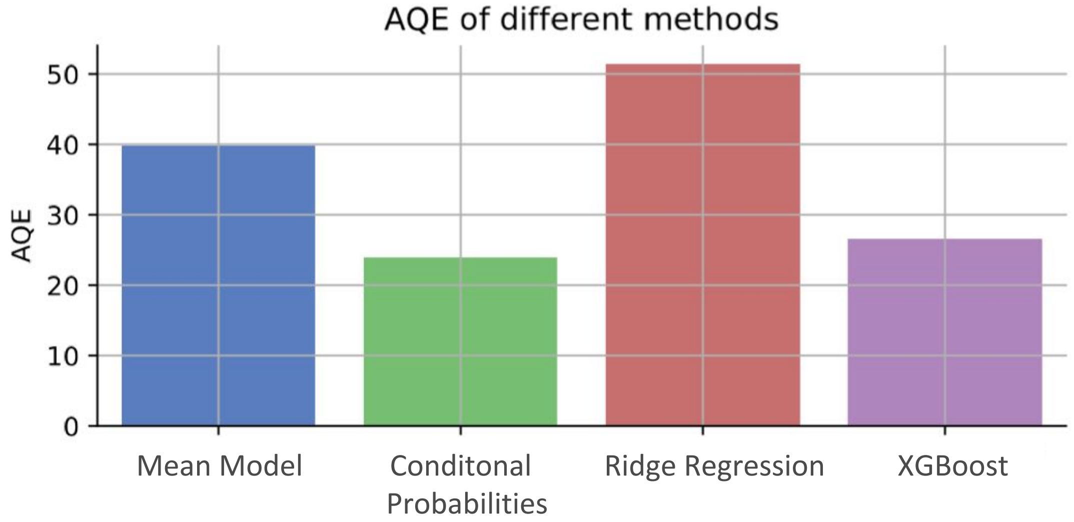
- Train separate model for each user
- Nature of data invalidates cross-validation assumptions

➔ Rolling-origin update evaluation:



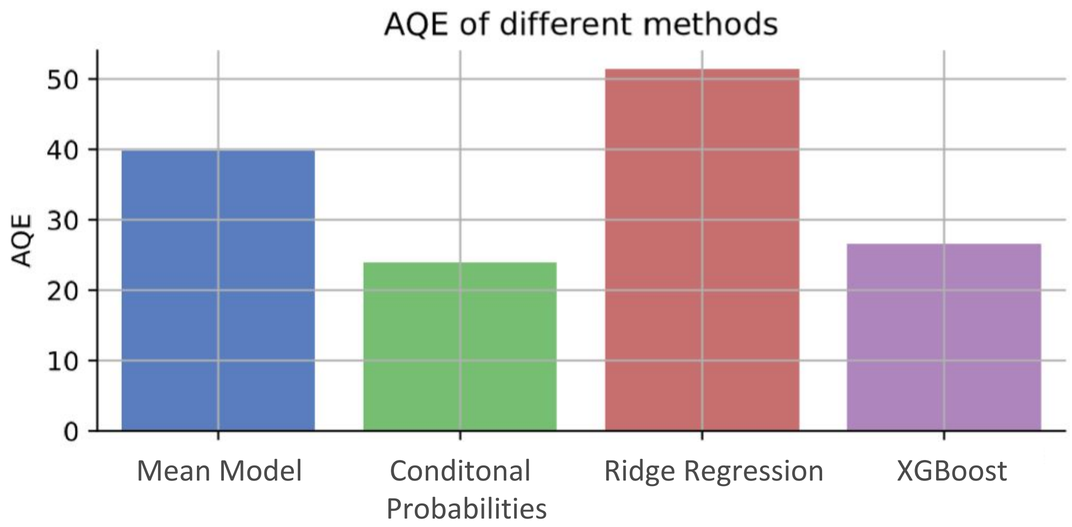
Results for methods trained on single users

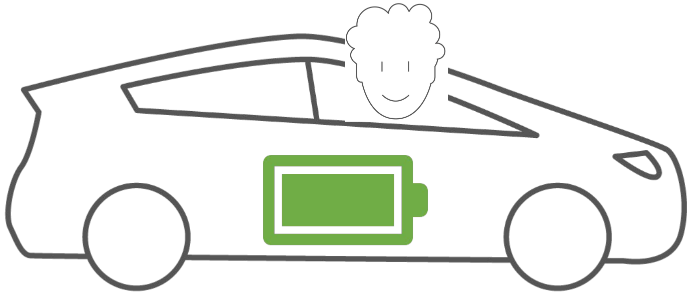
MAE: 2,55 2,98 2,65 2,43



Results for methods trained on single users

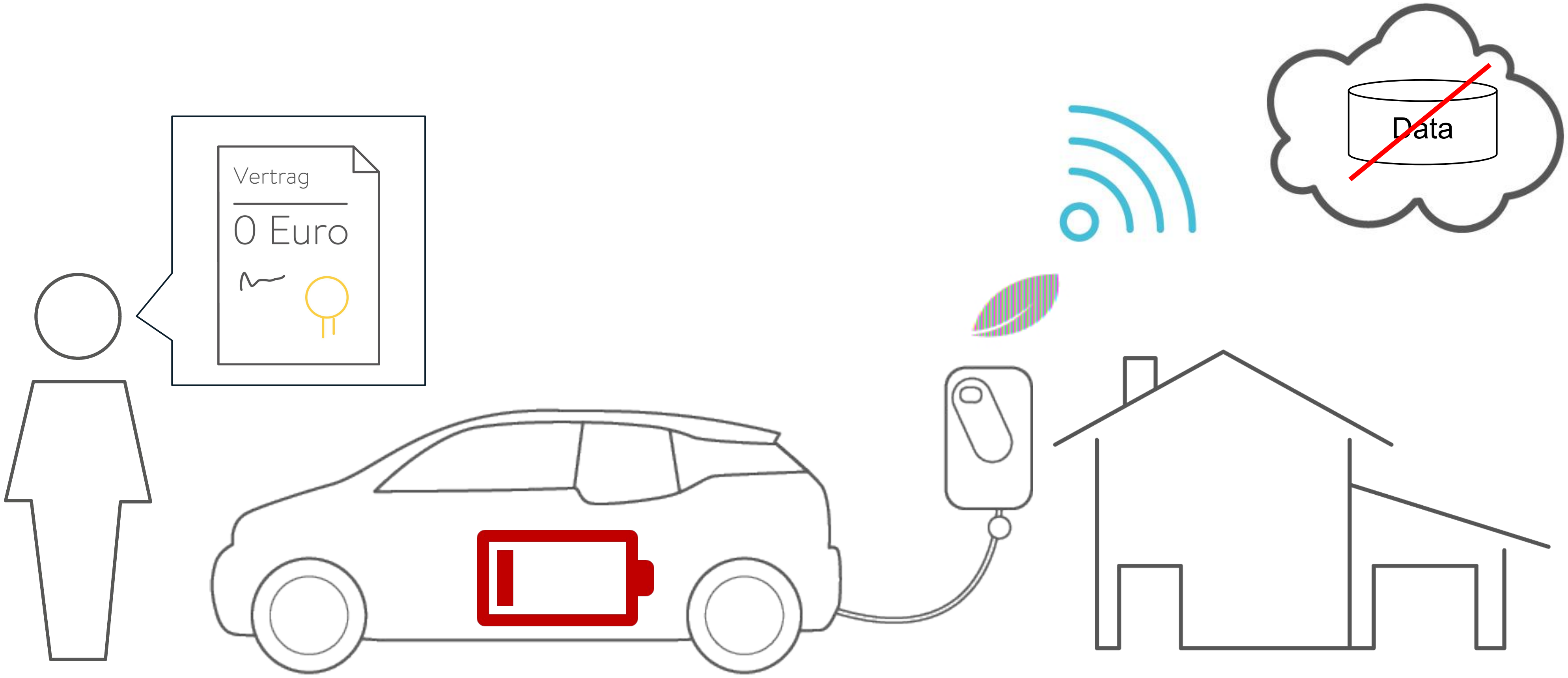
MAE: 2,55 2,98 2,65 2,43





Scenario 3

New User

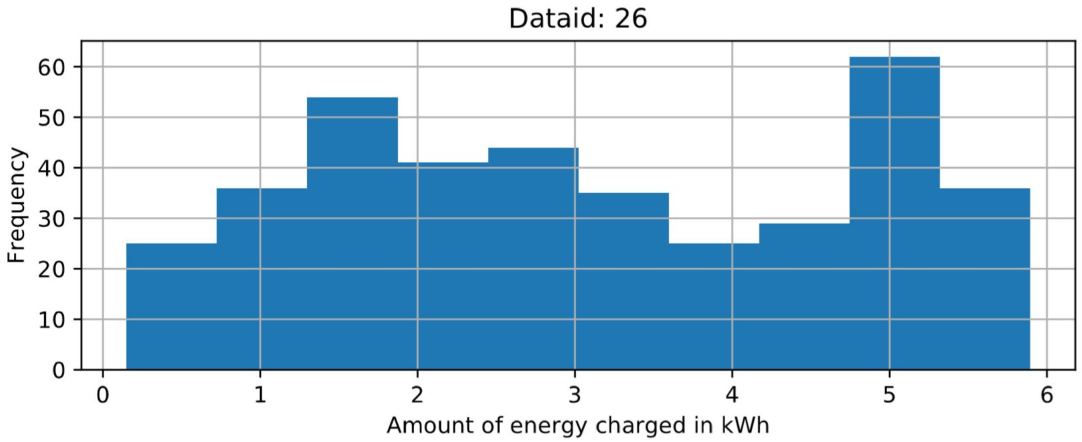


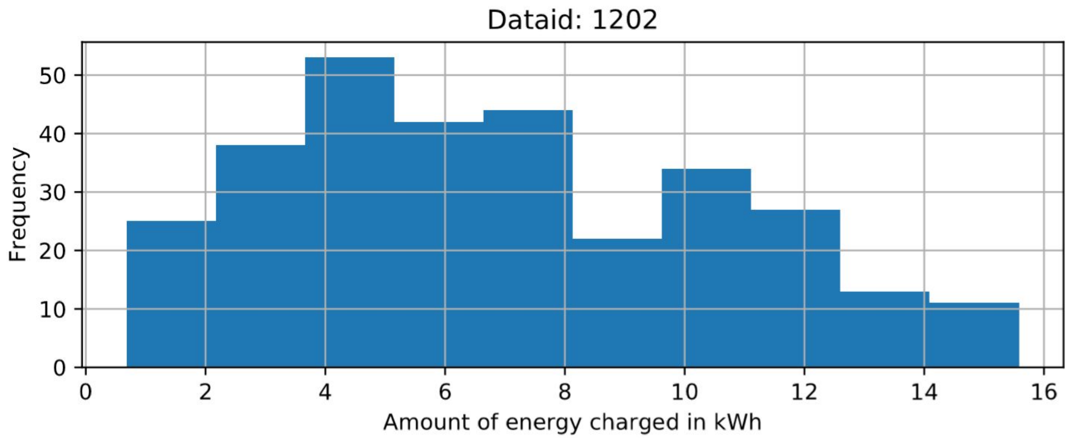
New Approach

Train on all available data

- Use data from all users for training

- However: great discrepancy in amount of energy charged:

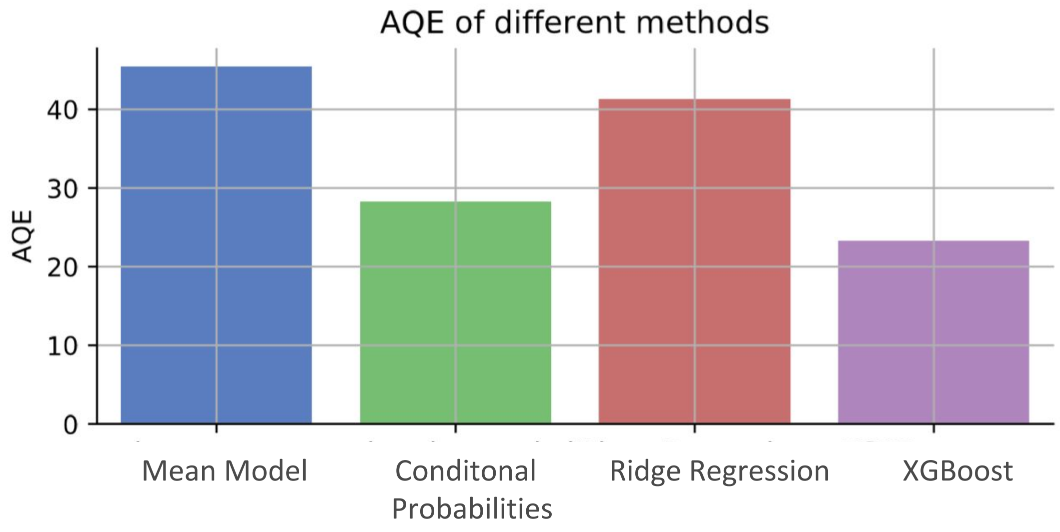




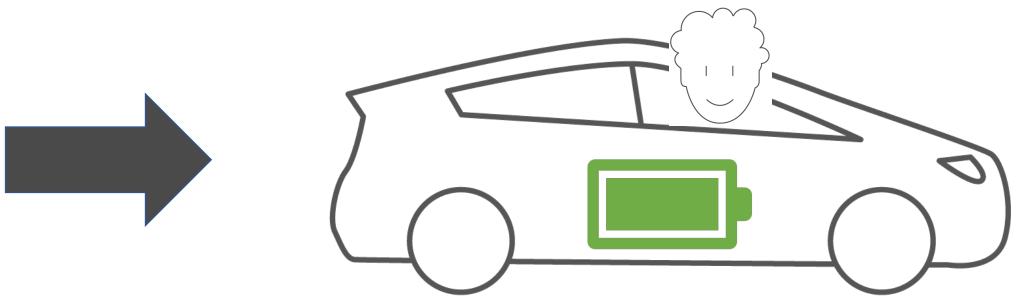
➔ Cluster users based on their (estimated) battery size

Results for methods trained on similar users

MAE: 3,05 3,51 2,47 2,95

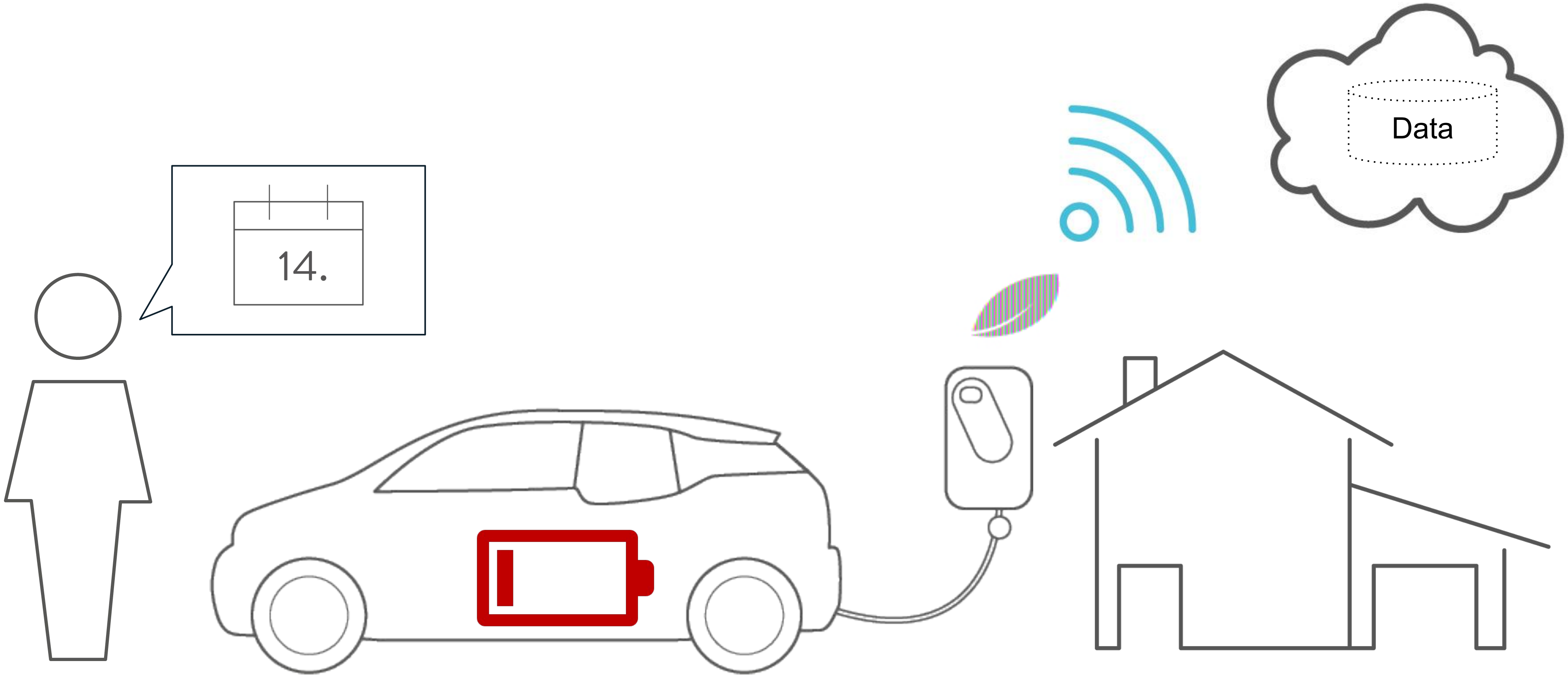


Results for methods trained on similar users



Scenario 4

Recent User

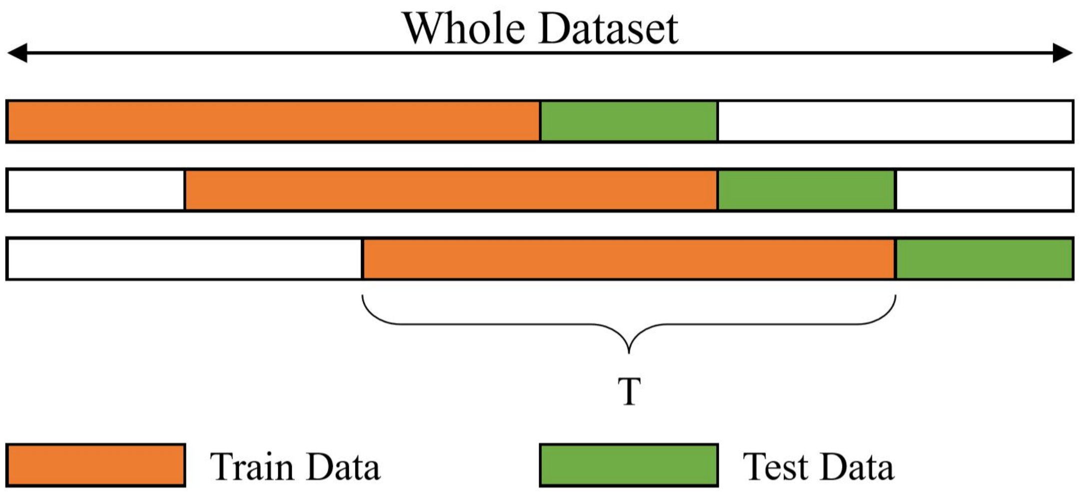


Rolling-window evaluation

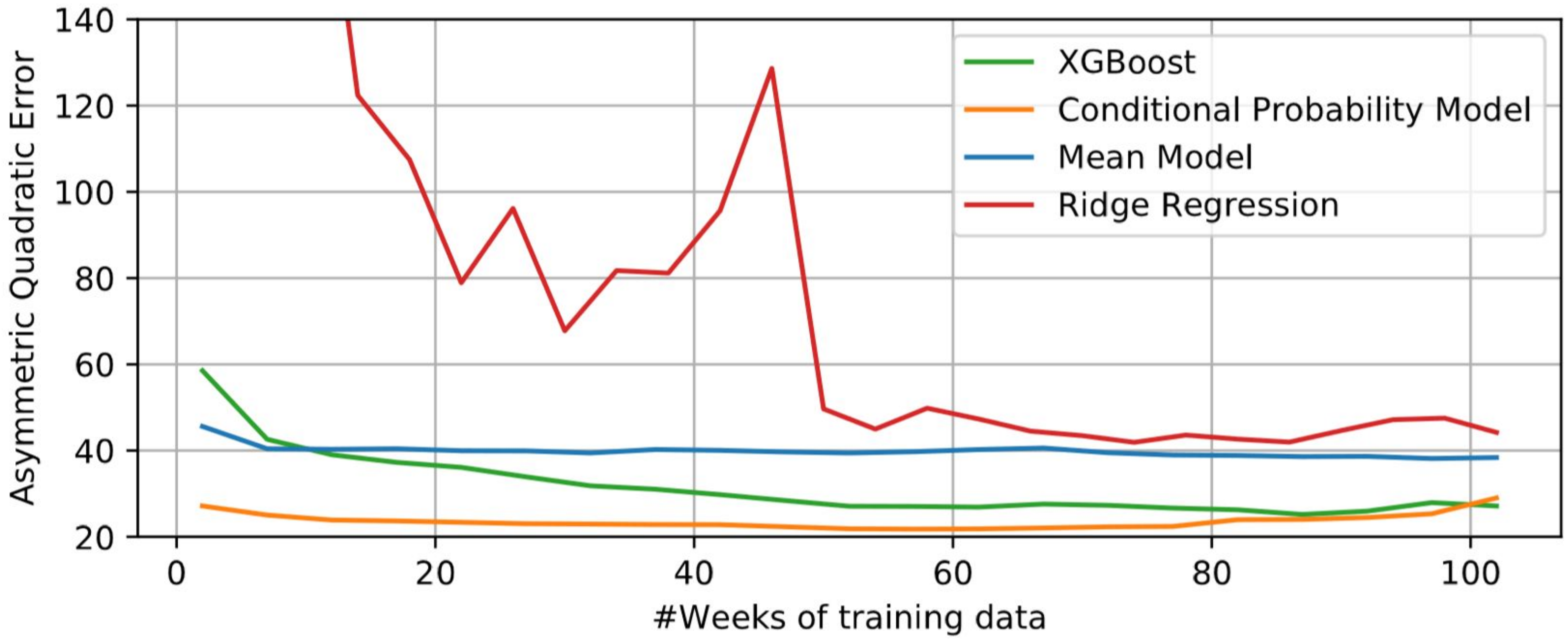
Evaluate performance of models on different amount of training data:

$\longrightarrow$ Rolling-window evaluation

For T = {2 weeks, 3 weeks, ..., 104 weeks}:

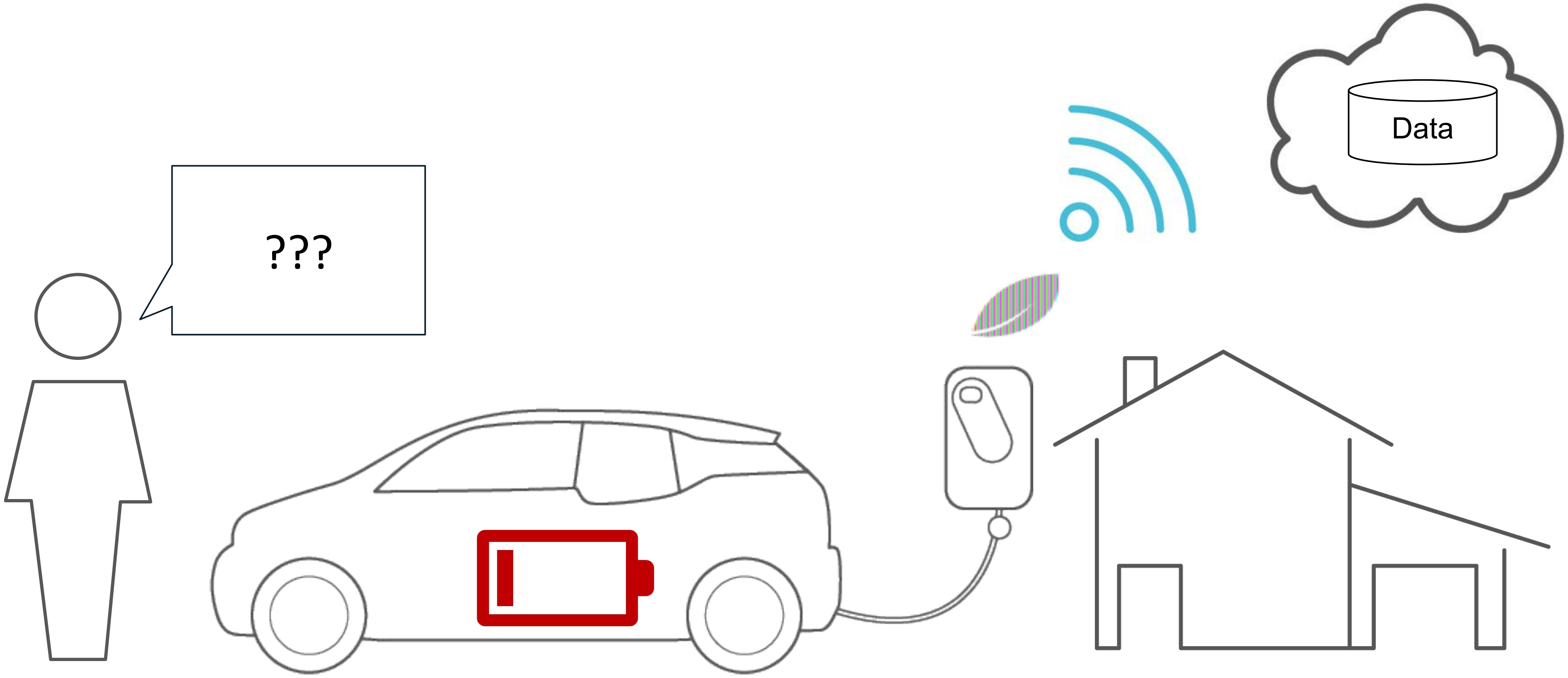


Results for different methods



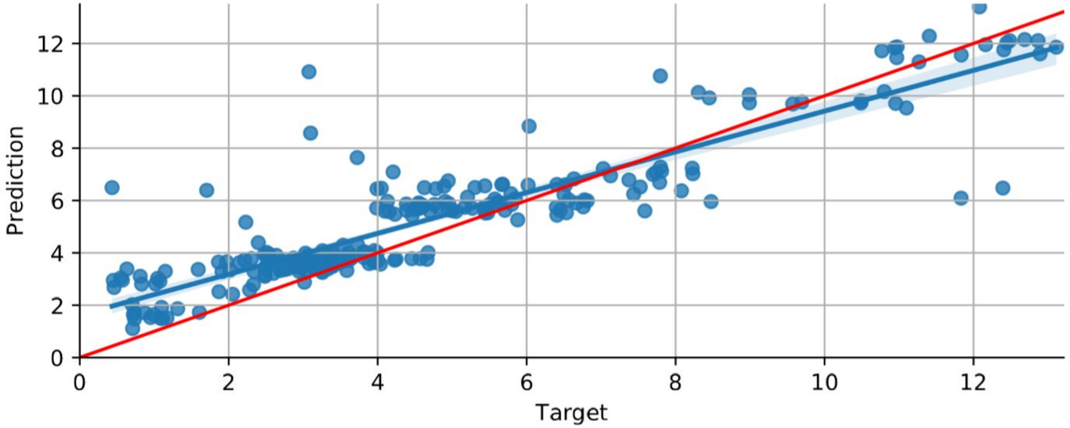
Scenario 5

Different user behaviors

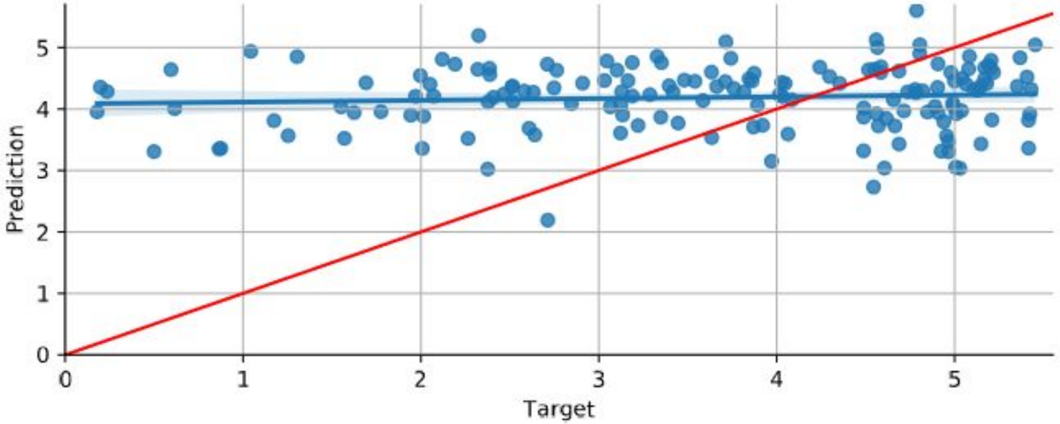


Results for different users using XGBoost

Prediction quality differs highly among different users:

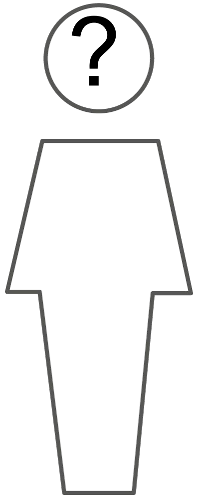


User 26



User 114

Correlations with AQE

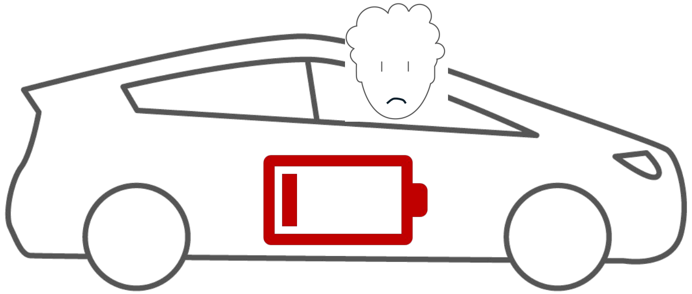


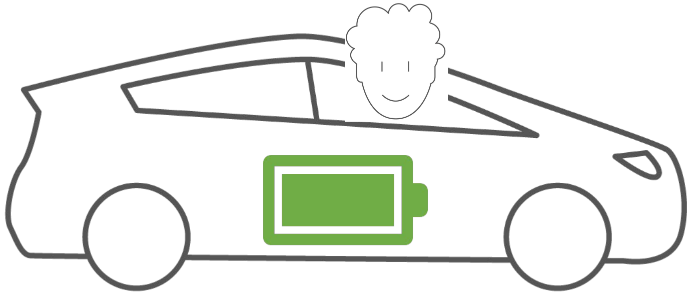
Feature	Correlation with AQE
Battery Size	-0.349
Charging Frequency	0.131
#Events	-0.369

Pearson Correlation Coefficients

Results of tests on different users

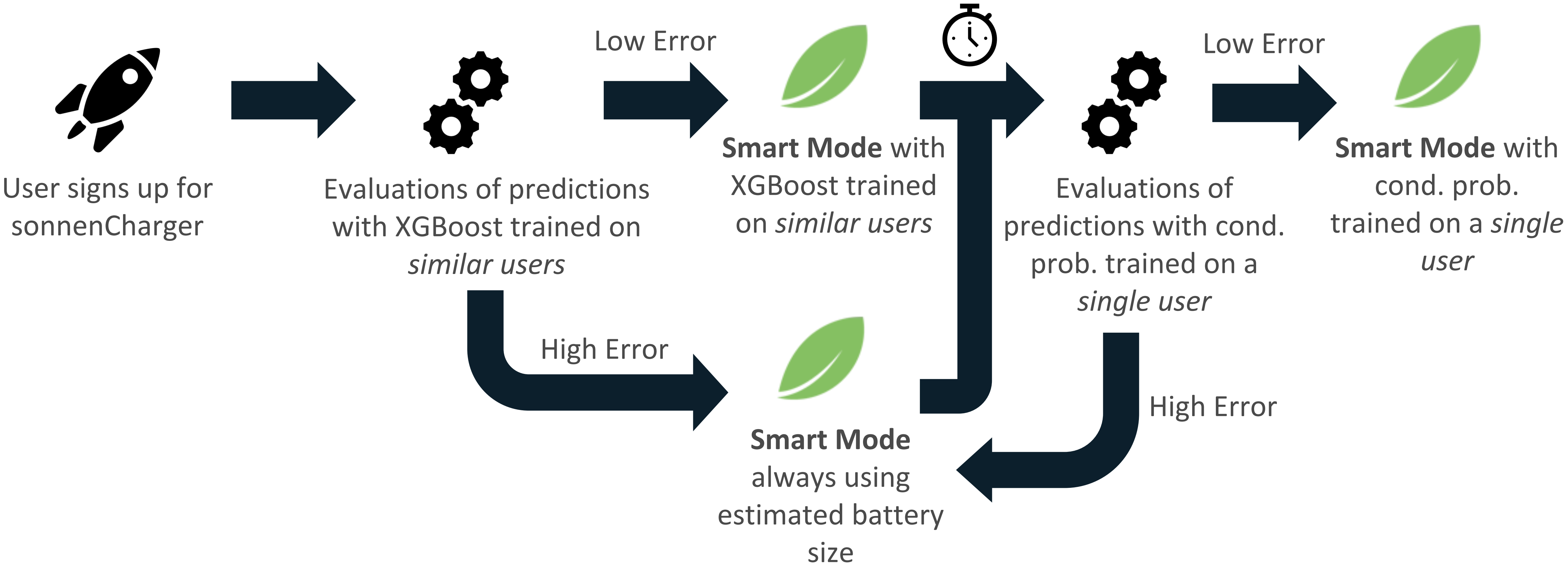
→ Methods are only applicable to some users:





Conclusion

Final Approach



Final Approach

- Extension using convex combination of predictions and battery capacity:

$$y_{final} = \alpha \cdot y_{pred} + (1 - \alpha) \cdot capacity_{est}$$

- Choice of α:
  - Inversely proportional to AQE
  - Based on user preferences

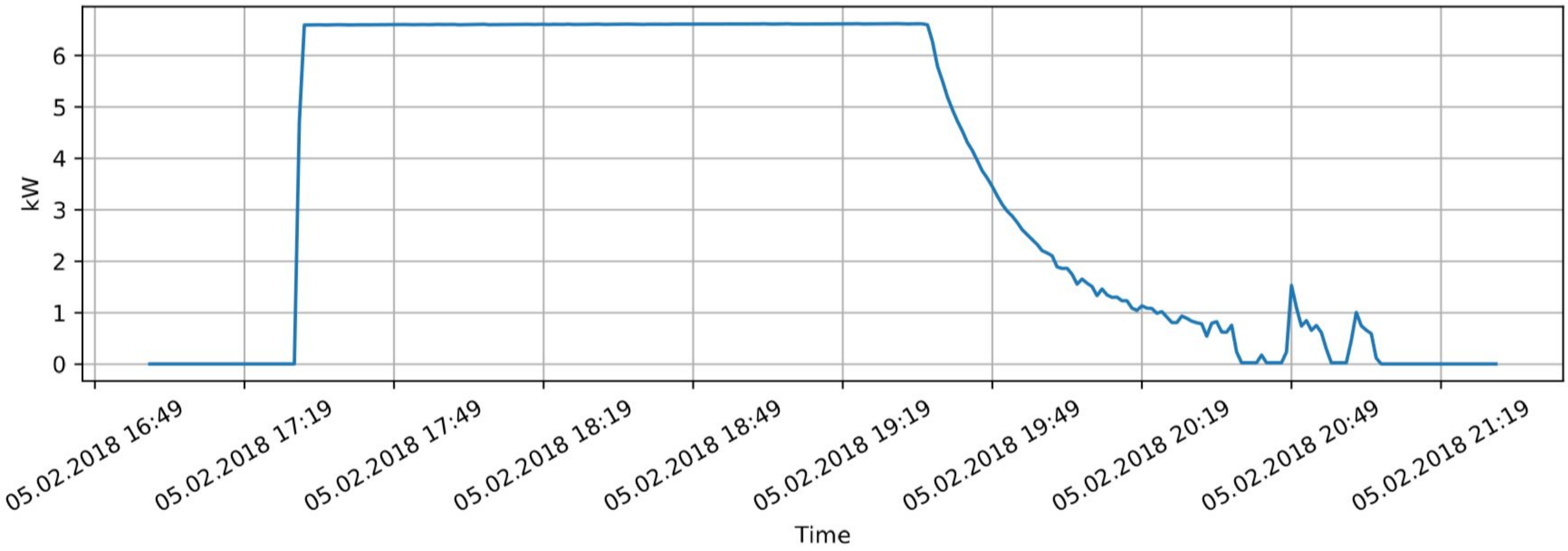
Q&A

sonnen

TUM Data Innovation Lab

- Projects proposed by companies
- Interdisciplinary student teams
- Data-driven solutions

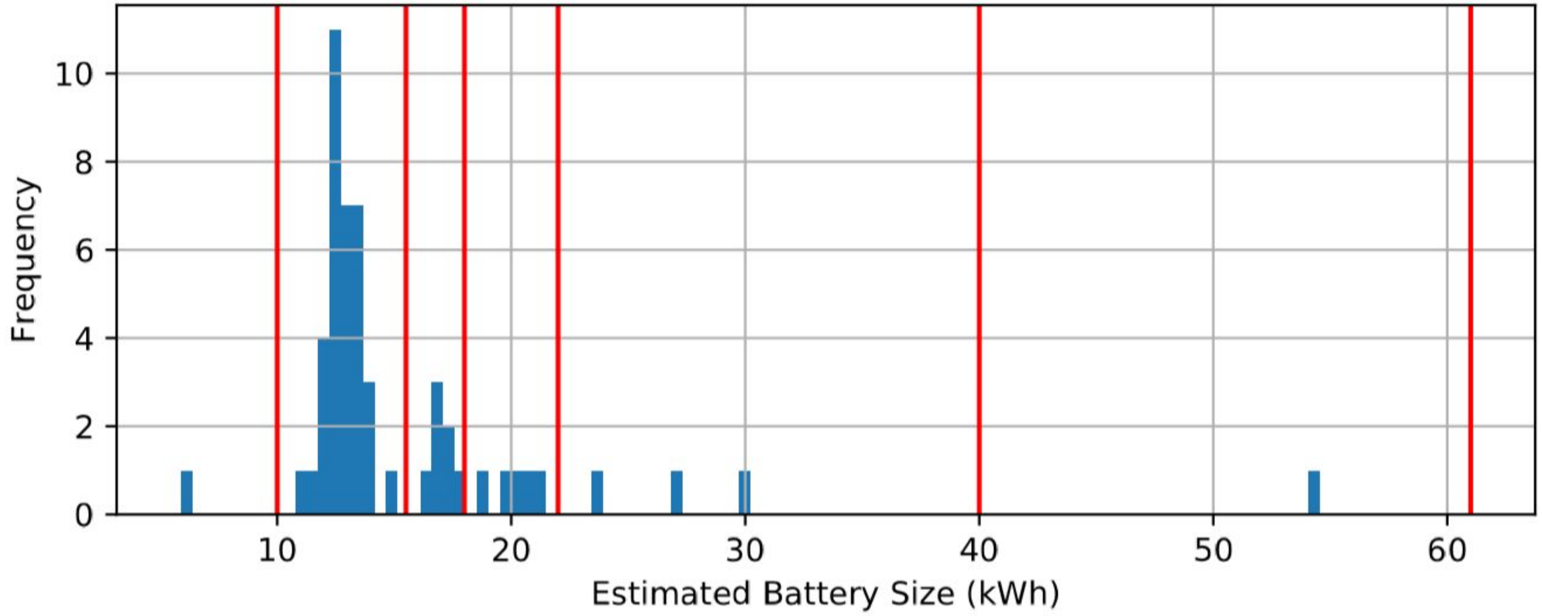
https://www.di-lab.tum.de/



kW
0
1
2
3
4
5
6
05.02.2018 16:49
05.02.2018 17:19
05.02.2018 17:49
05.02.2018 18:19
05.02.2018 18:49
05.02.2018 19:19
05.02.2018 19:49
05.02.2018 20:19
05.02.2018 20:49
05.02.2018 21:19
Time

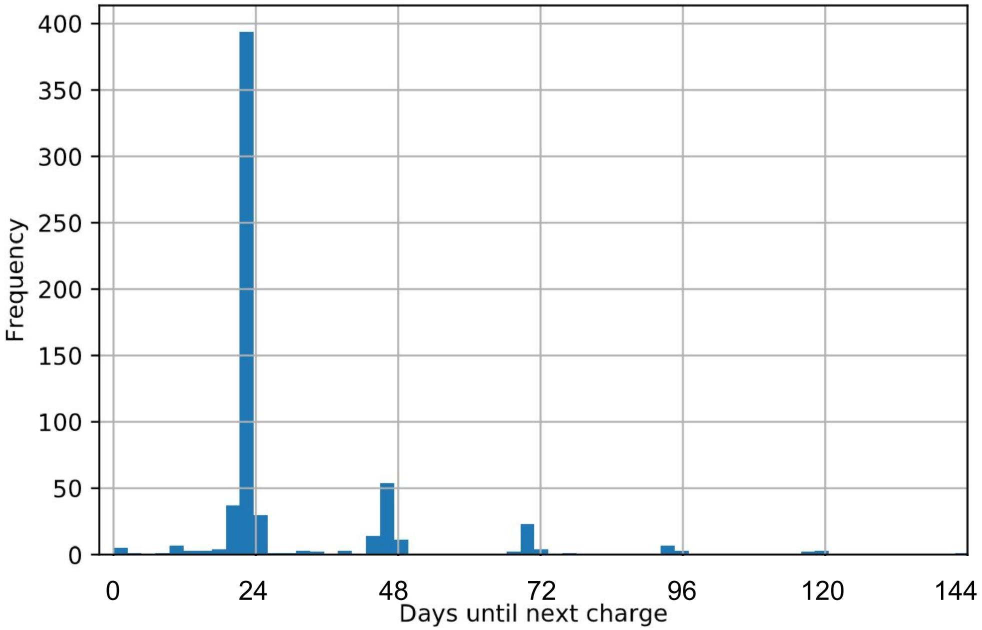
Train only on similar users

Cluster users based on their (estimated) battery size:

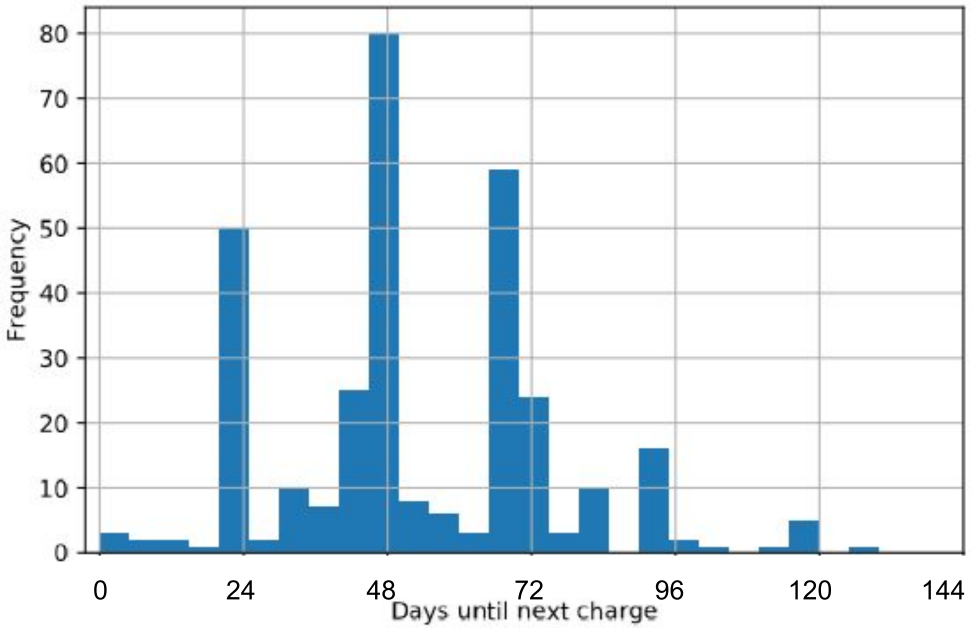


Delay between charges

How often do users charge?

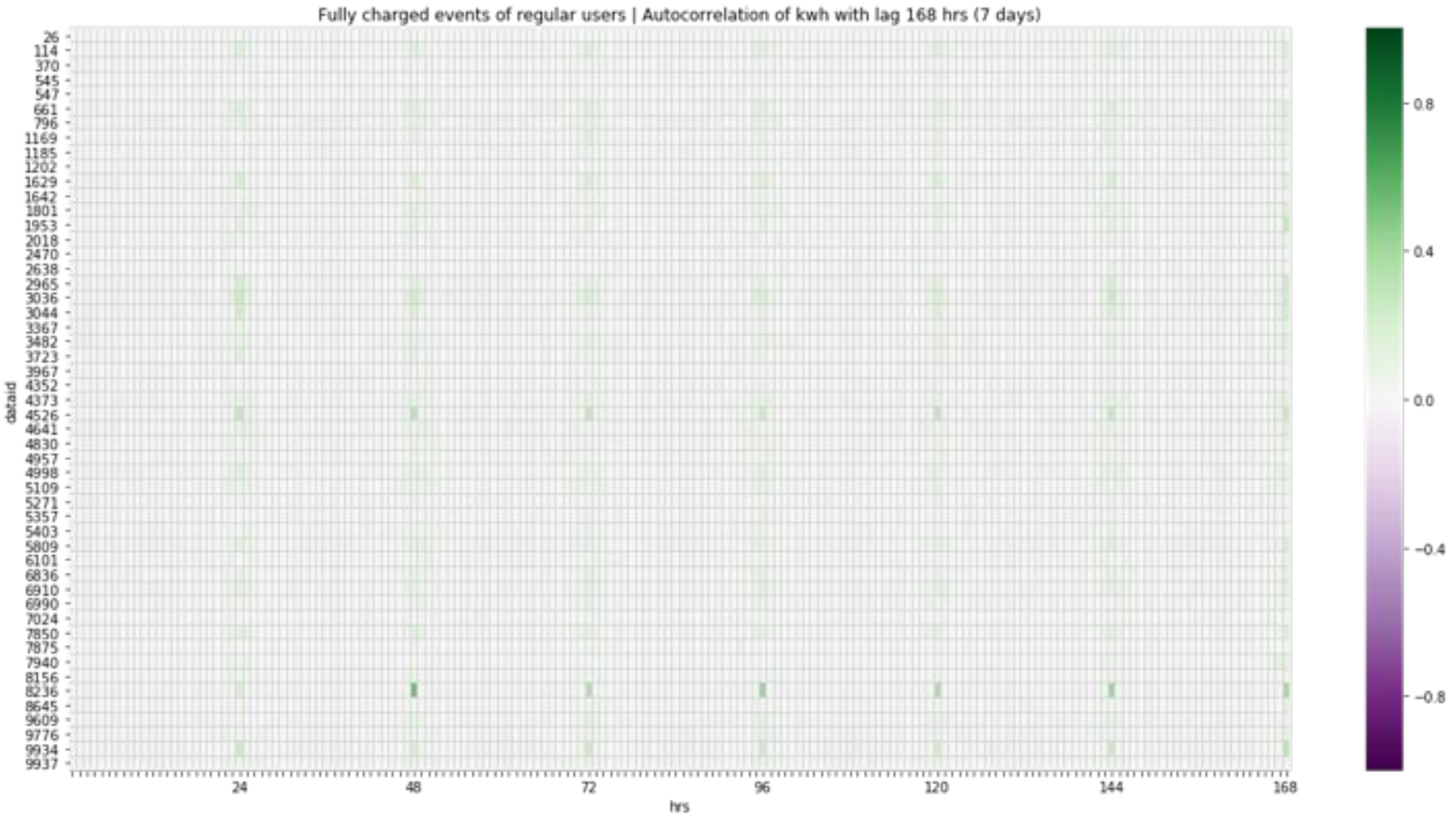


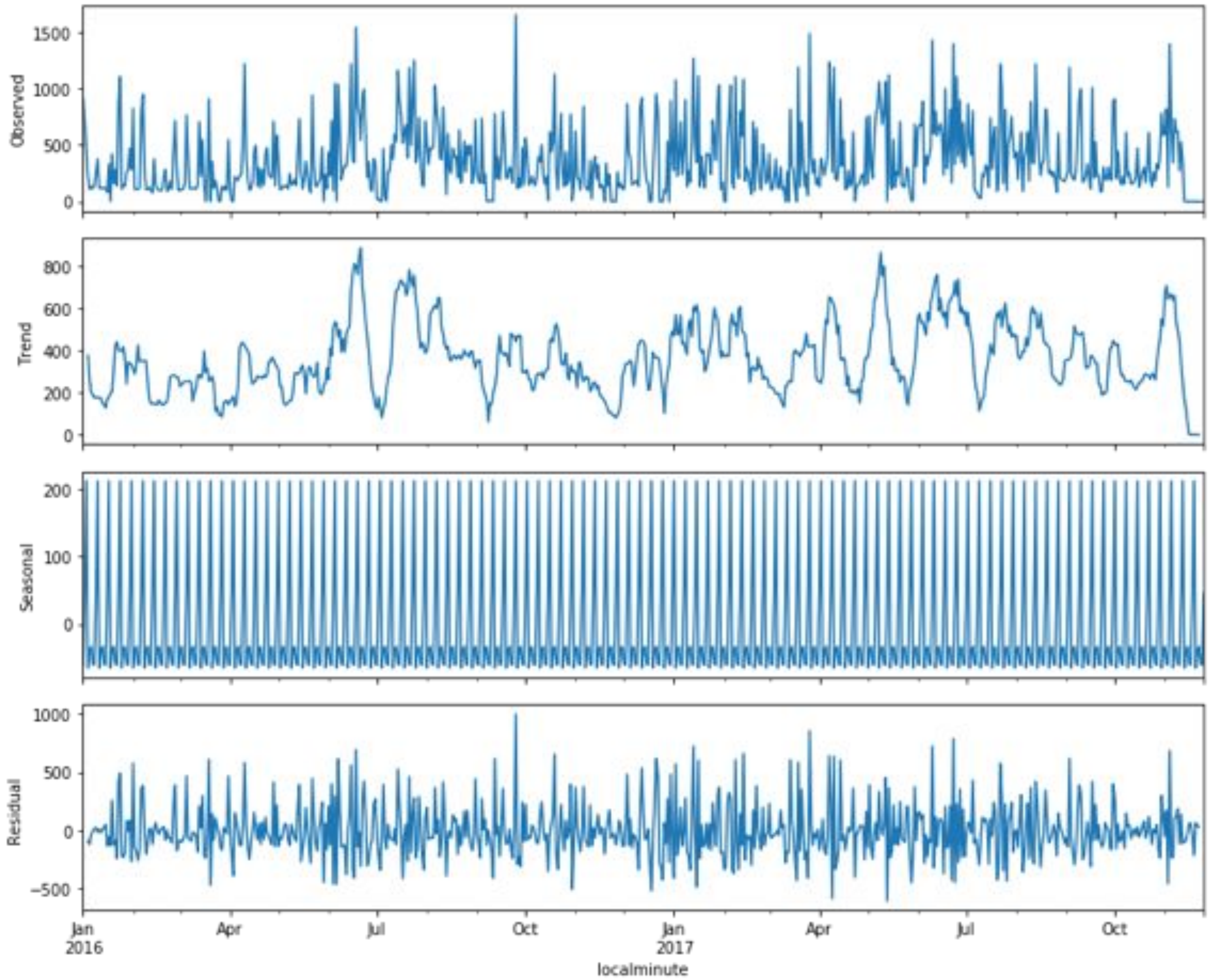
User 114



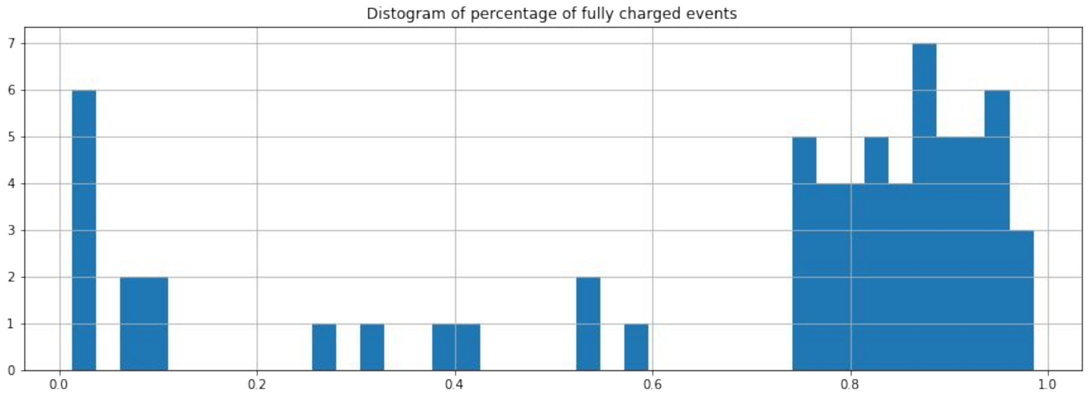
User 1169

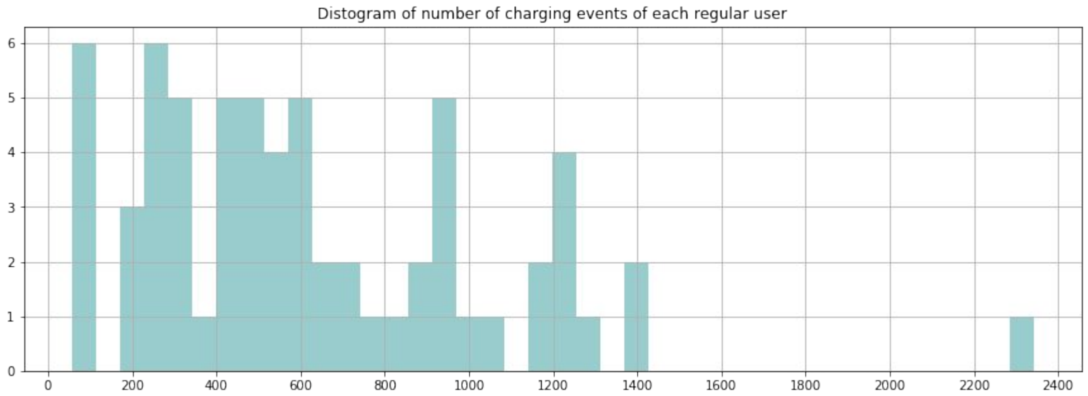
→ Very regular timespan in between charges


Fully charged events of regular users | Autocorrelation of kwh with lag 168 hrs (7 days)
dataid
26
114
370
545
547
661
796
1169
1185
1202
1629
1642
1801
1953
2018
2470
2638
2965
3036
3044
3367
3482
3723
3967
4352
4373
4526
4641
4830
4957
4998
5109
5271
5357
5403
5809
6101
6836
6910
6990
7024
7850
7875
7940
8156
8236
8645
9609
9776
9934
9937
24
48
72
96
120
144
168
hrs
0.8
0.4
0.0
−0.4
−0.8


Observed
Trend
Seasonal
Residual
1500
1000
500
0
800
600
400
200
0
200
100
0
1000
500
0
−500
Jan
2016
Apr
Jul
Oct
Jan
2017
Apr
Jul
Oct
localminute

Data description





Data collected from 2016.01.06 to 2018.05.01

- 38580 charging events to 63 regular users
- 28730 fully charged
- 612 number of charging event per regular user

A regular user would

- charge EV 69% of the times to full,
- demand 7.85 kwh,
- 27% charging events occurs on weekend
- Of fully charged events, a regular user starts with 53% SOC